

Received 23 August 2023, accepted 22 September 2023, date of publication 25 September 2023, date of current version 4 October 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3319268



RESEARCH ARTICLE

ADOM: ADMM-Based Optimization Model for Stripe Noise Removal in Remote Sensing Image

ABSTRACT Remote sensing images (RSI) are useful for various tasks such as Earth observation and climate change. However, RSI may suffer from stripe noise due to physical limitations in sensor systems. Therefore, image destriping is essential, since stripe noise may cause serious problems in real applications. In this paper, we shall present a new Alternating Direction method of multipliers (ADMM)-based Optimization Model, called ADOM for stripe noise removal in RSI. First, we formulate an optimization function for finding stripe noise components from the observed image for stripe noise removal, and then optimization process for solving the optimization function in order to extract stripe noise component. In the optimization process, we shall propose a weight-based detection strategy for efficient stripe noise component capture, and an ADMM-based acceleration strategy for fast stripe noise removal. In the weight-based detection strategy, we effectively detect stripe noise similar to the image details by using weighted norm generated by adjusting norm and group norm weights based on the momentum coefficient and residual parameter. In the ADMM-based acceleration strategy, we accelerate optimization process by using two control strategies: evidence-based starting point control and momentum-based step-size control. The former provides a starting point for more accurately finding stripe noise component, and the latter accelerates convergence by using the momentum coefficient while providing optimization stability by exploiting the damping coefficient. Our experimental results show that ADOM achieves better performance for both of simulated and real image data sets compared to the other destriping models.

INDEX TERMS Destriping, remote sensing image, alternating direction method of multipliers, optimization.

I. INTRODUCTION

Remote sensing images (RSI) acquired from satellites or airborne platforms provide rich information for decision-making in various fields such as monitoring, deforestation, environmental measurements, weather observations, early warning and natural disaster prediction [1], [2], [3], [4], [5], [6]. Calibration plays a crucial role in the analysis of RSI by adjusting the sensor measurement values into the corresponding actual ground truth values [7]. However, stripe noise often occurs in RSI after calibration due to the physical limitations in sensor systems with a gap between

The associate editor coordinating the review of this manuscript and approving it for publication was Ikramullah Lali.

detectors in relative gain and offset [8], [9]. The stripe noise in an image can have a negative impact on visual quality, and may cause serious problems in applications such as classification and object detection [10], [11], [12], [13], [14], [15]. Since the hardware upgrade may not completely solve this problem, it is crucial and essential to develop image processing techniques for strip noise removal in RSI.

Image destriping refers to the process of removing stripes from an image. It is an active research topic in the field of image processing. Over the past few decades, the numerous image destriping methods are developed for RSI. They can be roughly classified into four categories: filtering-based methods, statistical-based methods, deep learning-based methods and optimization-based methods. Filtering-based

methods are efficient and easy to implement, but some results can suffer from blurring artifacts due to excessive filtering [8], [16], [17], [18], [19], [20], [21], [22], [23], [24]. On the other hand, statistical-based methods quickly remove stripe noise by exploiting the statistical properties of images. However, their performance is often limited, as they rely on reference images [25], [26], [27], [28], [29], [30]. Deep learning-based methods have emerged as a popular solution for stripe noise removal owing to their ability to effectively extract key features from images [31], [32], [33], [34], [35], [36], [37], [38], [39], [40], [41], [42], [43], [44], [45]. However, the performance of such methods is limited by the domain-dependency problem, which is influenced by the quantity and quality of the training data, and restricts their generalization ability. Despite the efforts for reducing the impact of domain dependencies, improving the performance of deep learning models is still challenging due to a lack of reliable ground truth data. In addition, the development of deep learning models requires a lot of experiments, which often involve significant computational resources, resulting in the increase of carbon emissions [46]. In other words, deep learning models can have a negative impact on the environment, due to the significant computational resources required for training and the resulting increase in carbon emissions. In recent years, optimization-based methods have shown promising results in the field of image despeckling [9], [47], [48], [49], [50], [51], [52], [53], [54], [55], [56], [57], [58], [59], [60], [61], [62], [63], [64], [65], [66], [67], [68], [69], [70], [71], [72], [73], [74], [75], [76], [77], [78]. These methods incorporate prior knowledge of the image or stripe noise into the objective function in order to effectively remove the stripe noise. An advantage of these methods over deep learning-based methods is their ability to perform stripe noise removal without requiring a large amount of training data. However, the existing optimization-based methods still have limitations in two aspects.

First, they often lead to inaccurate stripe noise removal results due to the limitation of preserving image details, such as edges and textures similar to stripe noise component. To address this problem, some existing optimization-based methods have used functions such as L_0 , L_1 and $L_{2,1}$ -norm, which represent prior knowledge about the direction and structural features of stripes [48], [49], [50], [51], [52], [53], [54]. In addition, the weighted norm has also been used to effectively find stripes [55]. However, the existing methods tend to over-smooth the image, which leads to the loss of image details. Also, they have limitations in accurately finding stripe noise component. Therefore, a new weighted norm is required to effectively preserve the image details while improving stripe noise removal performance.

Second, the optimization process involves iteratively seeking the optimal solution until a satisfactory stopping criterion is reached, which can result in a significant computational overhead. Recently, the alternating direction method of multipliers (ADMM) has been widely used in image despeckling by decomposing the optimization problem

into smaller subproblems to efficiently derive the optimal solution for the objective function [79]. However, ADMM has the time-consuming problem, since the updates become less variable as the solution approaches to the convergence. Therefore, an accelerated method for ADMM is required to remove stripe noise from images very fast.

In this paper, we shall propose a new ADMM-based Optimization Model (ADOM) for stripe noise removal in RSI. While the existing optimization-based methods mainly focus on enhancing visual quality, we shall propose not only a weight-based detection strategy for efficient stripe noise component capture, but also an ADMM-based acceleration strategy for fast stripe noise removal. Specifically, in the weight-based detection strategy, we effectively detect stripe noise similar to the image details by using weighted norm, which is generated by adjusting norm and group norm weights based on the momentum coefficient and residual parameter. In the ADMM-based acceleration strategy, we accelerate optimization process by using two control strategies: evidence-based starting point control and momentum-based step-size control. The former provides a starting point by adjusting the threshold parameter properly in order to more accurately find stripe noise component, and the latter accelerates convergence by using the momentum coefficient while providing optimization stability by exploiting the damping coefficient.

In summary, our main contributions can be highlighted by the following five points.

- 1) We propose ADOM for stripe noise removal in RSI. For this purpose, we formulate an optimization function for finding stripe noise components from the observed image for stripe noise removal, and then optimization process for extracting stripe noise component by solving the optimization function.
- 2) We propose a weight-based detection strategy for efficient stripe noise component capture and an ADMM-based acceleration strategy for fast stripe noise removal.
- 3) In the weight-based detection strategy, we effectively detect stripe noise by using weighted norm generated by adjusting norm and group norm weights.
- 4) In the ADMM-based accelerated strategy, we accelerate optimization process by using two control strategies: evidence-based starting point control and momentum-based step-size control.
- 5) We demonstrate that our ADOM achieves better performance compared to the other despeckling methods through extensive experiments.

The remainder of this paper is outlined as follows: Section II presents related work to our approach. Some important mathematical symbols and terms are explained in Section III. The main structure of our proposed ADOM is explained in Section IV. The overall experimental results and corresponding analysis are presented in Section V. Finally, our conclusions are presented in Section VI.

II. RELATED WORK

In this section, we shall briefly explain about four different categories of image destriping methods: filtering-based methods, statistical-based methods, deep learning-based methods and optimization-based methods.

A. FILTERING-BASED METHODS

Filtering-based methods remove stripe noise in the spatial or transformed domains. They commonly use filtering techniques such as Fourier transform and wavelet decomposition. Fourier transform-based methods remove stripe noise in the frequency domain [8], [16], [17]. In contrast, wavelet decomposition-based methods remove stripe noise using directional properties [18], [19]. Wavelet-Fourier transform-based methods have been presented in order to improve stripe noise removal performance [20], [21]. Hybrid approaches combining statistical matching and filtering techniques have also been proposed [22], [23], [24]. However, some filtering-based methods often suffer from blurring artifacts due to excessive filtering.

B. STATISTICAL-BASED METHODS

Statistical-based methods rely on the assumption that stripe noise has different statistical properties from those of the original image. Histogram matching and moment matching are two commonly used methods. The former focuses on adjusting the histogram of a noisy image to match that of a reference image [25], [26], [27], whereas the latter adjusts the moments of a noisy image to match those of a reference image [28], [29], [30]. Statistical-based methods quickly remove stripe noise using the statistical properties of images, but their performance is often limited, as they rely on reference images.

C. DEEP LEARNING-BASED METHODS

Recently, deep learning-based methods have made significant progress in the field of image destriping due to their ability in order to automatically extract desired features from images. In particular, convolutional neural networks (CNN) are widely used as the most popular deep learning method for image destriping [31], [32], [33], [34], [35], [36], [37], [38], [39], [40], [41], [42]. For example, Kuang et al. [31] first proposed a CNN model for stripe noise removal and super resolution. Xiao et al. [32] proposed a CNN model using residual learning for better performance and fast convergence. Chang et al. [33] presented a CNN model to remove stripe noise while preserving spectral-spatial information in a hyperspectral image (HSI) using residual learning, dilated convolution and multichannel filtering. SNRWDNN uses a wavelet transform to preserve the image details. SSGN utilizes multi-scale convolution blocks to minimize spectral distortion [35]. SSGN uses multi-scale convolutional blocks for stripe noise removal [37]. SGIDN utilizes 3D convolution, 2D convolution and residual learning for stripe noise removal [38]. Chang et al. [39] presented

a two-stream CNN model using wavelet transform and discriminative multitask learning for stripe noise removal. Sun et al. [40] proposed an MLR-DBPFN with a high-quality image fusion process for stripe noise removal. Cao et al. [41] presented a learning-based kernel refinement network to preserve image details. Guan et al. [42] presented a deep recurrent convolution neural network with a gated convolution strategy.

Most existing deep learning-based methods using CNN are based on supervised learning. Supervised learning requires a large number of clean and noisy image pairs. However, obtaining such data may be difficult due to time-consuming and expensive. To address this issue, Song et al. [43] presented an adversarial network by combining wavelet transform and CycleGAN for stripe noise removal. More recently, Wang et al. [44] presented a Translution-SNet by combining CNN and transformer to extract key features between stripe components and stripe-free components. Pan et al. [45] presented a wavelet learning framework with a gradient-aware frequency attention block. However, deep learning-based methods have the limitation of removing stripe noise due to the problem of interpretability. Therefore, we focus on utilizing prior knowledge of stripe noise while preserving the details of the image to obtain a high-quality destriped image.

D. OPTIMIZATION-BASED METHODS

Optimization-based methods for stripe noise removal can be roughly divided into three categories: gradient-based methods, low-rank matrix recovery-based methods and low-rank tensor approximation-based methods. Gradient-based methods use the gradient information of an image [9], [47], [48], [49], [50], [51], [52], [53], [54], [55], [56]. For example, Shen et al. [47] proposed a maximum *a posteriori* framework using a prior likelihood probability density function to preserve the image details. Unidirectional total variation (UTV)-based methods use the directional characteristics of stripe noise [48], [49]. Chang et al. [50] proposed an anisotropic spectral-spatial total variation model (ASSTV) using spectral and spatial information for stripe noise removal. Liu et al. [51] proposed a generic destriping model using a modified Hodrick-Prescott filter for efficient stripe noise removal. However, they often lead to inaccurate stripe noise removal due to the direct removal of stripe noise from the image. To overcome this problem, Liu et al. [52] presented a destriping model that estimated the stripe noise component from an input image. Chen et al. [53] proposed a group sparsity-based model using $L_{2,1}$ -norm to preserve image details. Dou et al. [54] presented a directional L_0 sparse model using the structural properties of the stripe noise. The reweighted block sparsity (RBS) model uses adaptive weight control to better detect and remove stripe noise [55]. Song et al. [56] presented an EM-based framework using a maximum likelihood estimator of stripe noise.

Low-rank matrix recovery-based methods have been proposed by assuming that the desired image features can

be found in a subspace of lower dimensionality than the full spectral band [57], [58], [59], [60], [61], [62], [63], [64], [65], [66], [67], [68]. GRLD uses graph regularization and low-rank representation to preserve the geometrical structure of an image [58]. Zhang et al. [59] presented a low-rank matrix recovery framework that utilized prior knowledge of noise and the GoDec algorithm to remove mixed noise. Chang et al. [60] proposed a low-rank-based decomposition model using the directional characteristics of the stripes in a subspace. Chang et al. [61] presented a transformed low-rank image decomposition framework for stripe noise removal. Cao et al. [62] presented a non-local total variation model using the non-local region information of images to remove stripe noise. Yang et al. [63] presented a model using unidirectional high-order total variation and the Schatten 1/2-norm to preserve the image details. Zhang et al. [64] presented a double low-rank model using the low-rank characteristic of the noise-free HSI and the low-rank structure of the stripe noise to obtain a clean HSI. Zhao et al. [65] proposed a destriping model using the Hodrick-Prescott decomposition to provide effective information for the optimization process. Naganuma et al. [66] presented a general destriping framework using a flatness constraint to capture the nature of stripe noise. Su et al. [67] presented a fast graph Laplacian regularizer using the low-rank property of the image to efficiently remove stripe noise. Xu et al. [68] presented a component pixel-aware destriping model using the iteratively reweighted nuclear norm to separate stripe noise from the image. However, low-rank matrix recovery-based methods often fail to preserve image details, since they do not consider the spatial and spectral information of HSI.

To address this problem, low-rank tensor approximation-based methods have emerged in recent years [69], [70], [71], [72], [73], [74], [75], [76], [77], [78]. Some existing methods use ranks along each tensor mode to construct the sparsity in a sophisticated manner [69], [70], [71]. Cao et al. [72] proposed a robust principal component analysis model to provide accurate correlations among different HSI bands. LRTDTV and SSTV-LRTF use direct tensor modeling techniques to preserve the spatial structure of HSI [73], [74]. LRTDGS uses weighted group sparsity and low-rank Tucker decomposition to preserve the spatial and spectral information of HSI [75]. WLRTR uses non-local similarity to preserve the inherent spectral-spatial correlation structure of the image [76]. Liu et al. [77] presented a tensor-based model using fibered rank, PnP-based regularization and three-directional randomized T-SVD to preserve the image details. ATVDLR uses a low N-tubal rank approximation to effectively remove stripe noise while preserving the spatial-spectral information of HSI [78]. However, existing optimization-based methods have limitations in efficiently removing stripe noise, since they focus on enhancing visual quality. Our ADOM employs not only a weight-based detection strategy for efficient stripe noise component capture, but also an ADMM-based acceleration strategy for fast stripe noise removal.

III. PRELIMINARIES

In this section, some important mathematical symbols and terms are defined to help understand the main content of this paper, and then the problem formulation for the image destriping is explained.

A. NOTATIONS

The following notations are used throughout this paper, and briefly explained in this section. $\mathbf{R}^{m \times n}$ denotes a set of all real matrices with dimensions $m \times n$, where m and n represent the number of rows and columns respectively. For matrix $X \in \mathbf{R}^{m \times n}$, the L_0 -norm represents the number of non-zero elements in the matrix. For matrix $X \in \mathbf{R}^{m \times n}$, $X^{[j]}$ denotes the grouping of X , where $[j] \in \{1, 2, \dots, n\}$ is an index set corresponding to the j th group, which consists of specific elements in X . For matrix $X \in \mathbf{R}^{m \times n}$, the L_1 , L_2 , and $L_{2,1}$ -norm are defined as $\|X\|_1 = \sum_{i=1}^m \sum_{j=1}^n |X_{i,j}|$, $\|X\|_2 = \sqrt{\sum_{i=1}^m \sum_{j=1}^n X_{i,j}^2}$, and $\|X\|_{2,1} = \left(\sum_{j=1}^n \left(\sum_{i=1}^m |X_{i,j}|^2 \right)^{1/2} \right)^2$ respectively. The transpose of a matrix is represented by the symbol T , and the iteration number is indicated by k . ∇_x and ∇_y represent the horizontal and vertical differentiation operators respectively. “Regularization” is a technique for finding the optimal solution by penalizing the optimization model.

In addition, some definitions for the terms used are listed as follows.

Definition 1 (Weighted L_1 -Norm): For matrix $X \in \mathbf{R}^{m \times n}$, the weighted L_1 -norm of X is defined as follows:

$$\|X\|_{w_n,1} = \sum_{i=1}^m \sum_{j=1}^n w_{n,i,j} |X_{i,j}|, \quad (1)$$

where $w_n \in \mathbf{R}^{m \times n}$ denotes the norm weight. The weighted L_1 -norm is calculated by multiplying the absolute value of X by its corresponding norm weight and then summing all the results.

Definition 2 (Weighted L_2 -Norm): For matrix $X \in \mathbf{R}^{m \times n}$, the weighted $L_{2,1}$ -norm of X is defined as follows:

$$\|X\|_{w_g,1} = \sum_{j=1}^n w_{g,j} \|X^{[j]}\|_2, \quad (2)$$

where $w_g \in \mathbf{R}^{m \times n}$ is the group norm weight, and g_j denotes the group corresponding to the j th column. The weighted $L_{2,1}$ -norm is calculated by multiplying the L_2 -norm of each group by its corresponding group norm weight, and then summing these values.

Definition 3 (Sign Function): For scalar a , the sign function is defined as follows:

$$\text{sign}(a) = \begin{cases} 1, & \text{if } a > 0, \\ 0, & \text{if } a = 0, \\ -1, & \text{otherwise.} \end{cases} \quad (3)$$

This function returns 1 if a is positive, 0 if a is zero and -1 if a is negative.

B. PROBLEM FORMULATION

By assuming that striping artifacts are additive noise, the degradation process in RSI can be expressed as:

$$O = D + S, \quad (4)$$

where $O \in \mathbb{R}^{m \times n}$, $D \in \mathbb{R}^{m \times n}$ and $S \in \mathbb{R}^{m \times n}$ denote observed image, desired clear image and stripe noise component respectively. Equation (4) represents O as the sum of D and S , and it can be more practical for stripe noise removal due to its simplicity and computational efficiency despite the potential presence of other noise characteristics. To simplify the analysis of stripe noise, we assume that the stripe noise is distributed vertically, although it may also exist horizontally in O .

To effectively remove the stripe noise, we focus on finding S instead of directly removing stripe noise from O . Thus, the stripe noise removal process can be represented as follows.

$$D = O - S \quad (5)$$

Equation (5) describes the process of separating S from O and producing D as the output.

IV. ADOM

In this section, we shall present ADOM for stripe noise removal in RSI. As shown in Fig. 1, our model can be described by optimization function and optimization process. First, we formulate an optimization function for finding stripe noise components from the observed image for stripe noise removal, and then explain about optimization process, which extracts stripe noise component by solving the optimization function through iterative steps. We can obtain the destriped image by separating the extracted stripe noise component from the observed image. In the following two sections, we shall describe about optimization function and optimization process for ADOM in detail respectively.

A. OPTIMIZATION FUNCTION

In this section, we shall explain about optimization function for our model, that is, augmented Lagrangian function derived from objective and constrained objective functions, which shall be described respectively below.

1) OBJECTIVE FUNCTION

Our purpose is to separate S from O utilizing prior knowledge of stripe noise while preserving the image details in order to obtain a high-quality D . Our objective function is defined as follows:

$$\arg \min_S \|\nabla_y S\|_1 + \lambda_1 \|\nabla_y(O - S)\|_{w_n,1} + \lambda_2 \|S\|_{w_g,2,1}, \quad (6)$$

where λ_1 and λ_2 are two positive regularization parameters for balancing the constraints between different terms. The first term $\|\nabla_y S\|_1$ represents L_1 -norm for the vertical gradient of S . As shown in Fig. 2, since the stripe noise does not appear in the vertical gradient of O , we can find S by minimizing the

vertical gradient of S . The second term $\lambda_1 \|\nabla_y(O - S)\|_{w_n,1}$ denotes the weighted L_1 -norm for the horizontal gradient of the difference between O and S . Since the stripe noise is mainly observed in the horizontal gradient of O as shown in Fig. 2, we can find S by minimizing the horizontal gradient of the difference between O and S . However, it may be difficult to find S accurately due to the similarity between some features of stripe noise and the image details. Existing methods using the traditional L_1 -norm may generate a loss of some information due to its tendency for excessively sticking to features similar to stripe noise [48]. To solve this problem, we use weighted L_1 -norm, which combines the traditional L_1 -norm with a norm weight w_n of equal intensity in order to detect S more accurately. The third term $\lambda_2 \|S\|_{w_g,2,1}$ represents the weighted $L_{2,1}$ -norm of S . The traditional $L_{2,1}$ -norm has limitation in finding stripe noise due to its sensitivity to outliers [53]. To address this problem, we use a weighted $L_{2,1}$ -norm, which combines the traditional $L_{2,1}$ -norm with a group norm weight w_g . We can detect S more effectively by using w_g in each group with different intensities at each iteration step, since weighted $L_{2,1}$ -norm is less sensitive to the outliers compared to the traditional $L_{2,1}$ -norm.

2) CONSTRAINED OBJECTIVE FUNCTION

We convert the objective function (6) into the following constrained objective function by introducing auxiliary variables A , B and C for computational efficiency.

$$\begin{aligned} & \arg \min_{A,B,C} \|A\|_1 + \lambda_1 \|B\|_{w_n,1} + \lambda_2 \|C\|_{w_g,2,1} \\ & \text{s.t. } A = \nabla_y S, B = \nabla_x(O - S), C = S \end{aligned} \quad (7)$$

3) AUGMENTED LAGRANGIAN FUNCTION

We derive the following augmented Lagrangian function from constrained objective function (7):

$$\begin{aligned} & \arg \min_{A,B,C,S} L_\rho(A, B, C, S, \tau_1, \tau_2, \tau_3) \\ & = \|A\|_1 + \tau_1^T (\nabla_y S - A) + \frac{\rho_1}{2} \|\nabla_y S - A\|^2 \\ & + \lambda_1 \|B\|_{w_n,1} + \tau_2^T (\nabla_x(O - S) - B) \\ & + \frac{\rho_2}{2} \|\nabla_x(O - S) - B\|^2 \\ & + \lambda_2 \|C\|_{w_g,2,1} + \tau_3^T (S - C) + \frac{\rho_3}{2} \|S - C\|^2, \end{aligned} \quad (8)$$

where τ_1 , τ_2 and τ_3 represent the Lagrange multipliers used to help convergence towards the optimal solution subject to the constraints, and ρ_1 , ρ_2 and ρ_3 are penalty parameters used to control the step-size of updates during the optimization process.

B. OPTIMIZATION PROCESS

In this section, we describe the optimization process for finding S , which satisfies the augmented Lagrangian function (8). The optimization process iteratively performs the

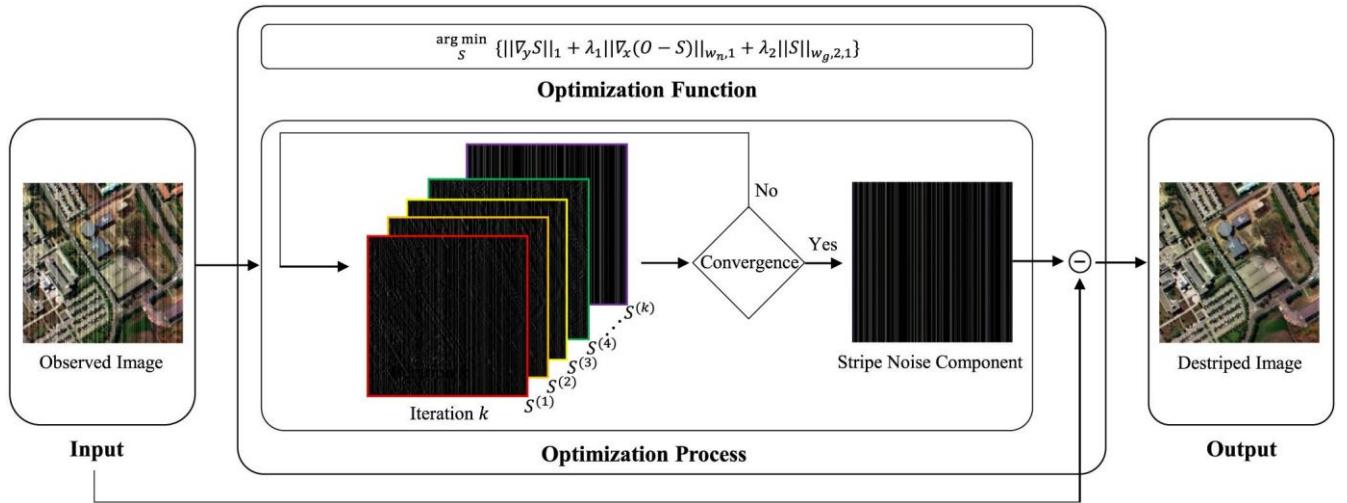


FIGURE 1. Overview of ADMM-based optimization model.

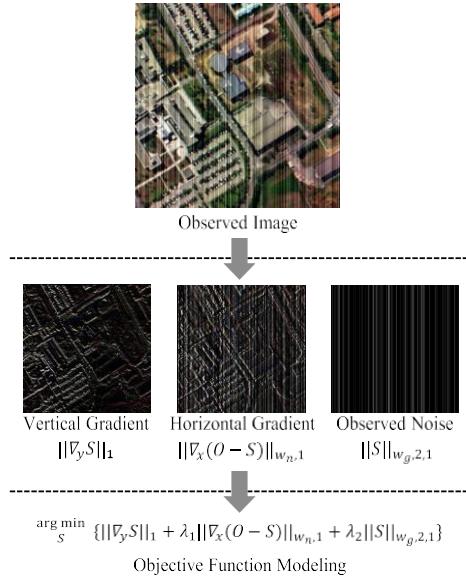


FIGURE 2. Objective function in ADOM.

following 4 steps: weight control, starting point control, step-size control and ADMM-based subproblem solving until convergence is achieved as shown in Fig. 3.

1) WEIGHT CONTROL

Weight control step updates norm and group norm weights by using the momentum coefficient and residual parameter in order to effectively detect stripe noise similar to the image details. First, we find a residual parameter γ by using the observed image O and stripe noise component S as follows:

$$\gamma^k = \frac{''(O - S^k) - (O - S^{k-1})''_2}{''(O - S^{k-1})''_2} \quad (9)$$

Then, we update norm weight w_n using γ as follows:

$$w_n^{k+1} = \frac{\alpha^{k-1} + \gamma^k}{\alpha^k - \gamma^k}, \quad (10)$$

where α is the momentum coefficient, and is initialized to 1 for optimization stability. Norm weight w_n improves stripe noise detection by properly adjusting the intensity according to the momentum coefficient α and residual parameter γ at each iteration step.

Finally, we update group norm weight w_g by using stripe noise component S and residual parameter γ as follows:

$$w_g^{k+1} = \begin{cases} \frac{1}{2} \frac{''S_g^k - S_g^k''_2}{2''S_g^k + \gamma^k''_2}, & \text{if } S_g^k - S_g^k < v_g^k, \\ w_g^k, & \text{otherwise.} \end{cases}$$

$$v_g^k = \frac{\sum_{j=1}^n S_g^k}{u_g}. \quad (11)$$

where, g_j is the group corresponding to the j th column of the image, and u_g denotes number of j th group elements. Group norm weight w_g improves stripe noise detection by using different intensities for each group at each iteration step.

We can effectively detect stripe noise similar to the image details by using weighted norm generated by adjusting norm and group norm weights based on the momentum coefficient and residual parameter.

2) EVIDENCE-BASED STARTING POINT CONTROL

Evidence-based starting point control step updates momentum coefficient α and damping coefficient d by adjusting the threshold parameter p properly in order to provide a starting point for more accurately finding stripe noise component. First, we update a momentum coefficient α , and then damping coefficient d using w_n and α . Momentum coefficient α is updated by using Nesterov's method in order to

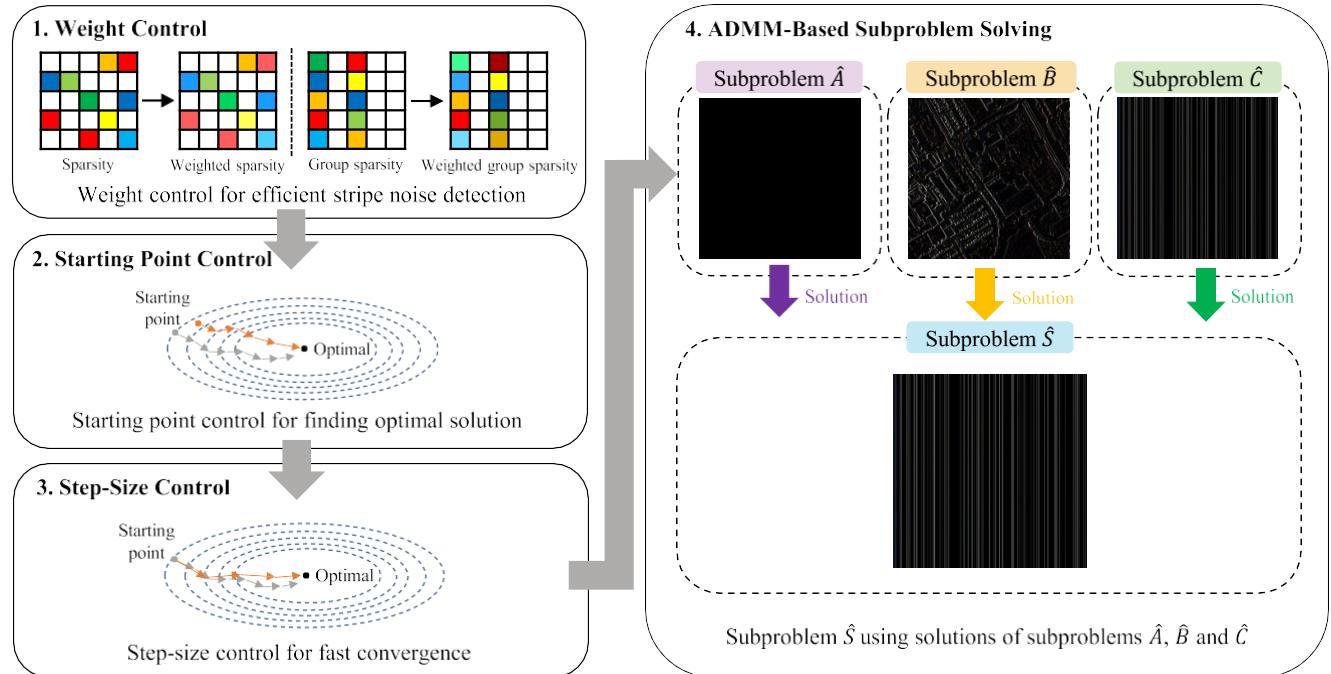


FIGURE 3. 4 steps for each iteration in optimization process.

provide fast convergence by properly adjusting the threshold parameter p [80].

$$\alpha^{k+1} = \begin{cases} \frac{\sqrt{1 + 4\alpha^k - 2}}{1 + 2\alpha^k}, & \text{if } k \leq p, \\ \frac{2}{1 + 2\alpha^k}, & \text{otherwise.} \end{cases} \quad (12)$$

Then, damping coefficient d is updated using w_n and α by properly adjusting the threshold parameter p in order to provide optimization stability by preventing excessive acceleration.

$$\alpha^{k+1} = \begin{cases} w_n^{k+1}, & \text{if } k \leq p, \\ \frac{\alpha^k}{\alpha^{k+1}}, & \text{otherwise.} \end{cases} \quad (13)$$

3) MOMENTUM-BASED STEP-SIZE CONTROL

Momentum-based step-size control step updates stripe noise component S and Lagrange multipliers τ_1 , τ_2 and τ_3 by using momentum coefficient α and damping coefficient d in order to accelerate convergence. First, stripe noise component S is updated by using momentum coefficient α for more fast convergence to S during subproblems solving as follows:

$$S^k = S^k + \frac{\alpha^{k+1} - \delta}{\alpha^k} S^k - S^{k-1} \quad (14)$$

, where δ is the constant, and set to 0.1 for fast convergence.

Then, Lagrange multipliers τ_1 , τ_2 and τ_3 are updated using damping coefficient d in order to provide optimization

stability by preventing excessive acceleration.

$$\begin{aligned} \tau_1^k &= d^{k+1} \tau_1^k, \\ \tau_2^k &= d^{k+1} \tau_2^k, \\ \tau_3^k &= d^{k+1} \tau_3^k. \end{aligned} \quad (15)$$

Momentum-based step-size control accelerates convergence by using the momentum coefficient α while providing optimization stability by exploiting the damping coefficient d .

4) ADMM-BASED SUBPROBLEM SOLVING

We solve our augmented Lagrangian function (8) by solving four subproblems \hat{A} , \hat{B} , \hat{C} and \hat{S} based on the ADMM [79].

$$\begin{aligned} \text{Subproblem } A : \text{Find } \hat{A} \text{ which minimize}_{\|A\|_1} & \tau_1^T (\nabla_y S - A) \\ & + \frac{\rho_1}{2} \|\nabla_y S - A\|^2. \end{aligned} \quad (16)$$

$$\begin{aligned} \hat{A} &= \arg \min_{\hat{A}} \left\{ \|A\|_1 + \frac{\rho_1}{2} \|\nabla_y S - A\|^2 \right\} \\ &= \arg \min_{\hat{A}} \left\{ \|A\|_1 + \frac{\rho_1}{2} \|\nabla_y S - A + \frac{\tau_1}{\rho_1}\|^2 \right\} \end{aligned}$$

$$\begin{aligned} \text{Subproblem } B : \text{Find } \hat{B} \text{ which minimize}_{\|B\|_{w_n,1}} & \lambda_1 \|B\|_{w_n,1} \\ & + \frac{\rho_2}{2} \|\nabla_x O - \nabla_x S - B\|^2. \end{aligned}$$

$$\begin{aligned} \hat{B} &= \arg \min_{\hat{B}} \left\{ \lambda_1 \|\hat{B}\|_{w_n,1} + \tau_2^T (\nabla_x O - \nabla_x S - B) \right. \\ &\quad \left. + \frac{\rho_2}{2} \|\nabla_x O - \nabla_x S - B\|^2 \right\} \\ &= \arg \min_{\hat{B}} \left\{ \lambda_1 \|\hat{B}\|_{w_n,1} \right\} \end{aligned}$$

$$+ \frac{\rho_2}{2} \|\nabla_x O - \nabla_x S - B + \frac{\tau_2}{\rho_2}\|_2^2 \} \quad (17)$$

$\tau^T \hat{S}_{Subproblem_C} + \frac{\rho_3}{2} \|S - C\|_2^2$. Find C which minimize $\{\lambda_2 \|C\|_{wg,2,1} +$

$$\begin{aligned} C^k &= \arg \min_C \{\lambda_2 \|C\|_{wg,2,1} + \tau_3^T (S - C) \\ &\quad + \frac{\rho_3}{2} \|S - C\|_2^2\} \\ &= \arg \min_C \{\lambda_2 \|C\|_{wg,2,1} \\ &\quad + \frac{\bar{\rho}_3}{2} \|S - C + \frac{\tau_3}{\rho_3}\|_2^2\} \end{aligned} \quad (18)$$

Subproblem \hat{S} : Find S which minimize
 $\{\tau_1^T (\nabla_y S - A) + \frac{\rho_1}{2} \|\nabla_y S - A\|_2^2 + \tau_2^T (\nabla_x O - \nabla_x S - B) + \frac{\rho_2}{2} \|\nabla_x O - \nabla_x S - B\|_2^2 + \tau_3^T (S - C) + \frac{\rho_3}{2} \|S - C\|_2^2\}.$

$$\begin{aligned} S^k &= \arg \min_S \{\tau_1^T (\nabla_y S - A) + \frac{\rho_1}{2} \|\nabla_y S - A\|_2^2 \\ &\quad + \tau_2^T (\nabla_x O - \nabla_x S - B) \\ &\quad + \frac{\rho_2}{2} \|\nabla_x O - \nabla_x S - B\|_2^2 \\ &\quad + \tau_3^T (S - C) + \frac{\rho_3}{2} \|S - C\|_2^2\} \\ &= \arg \min_S \{\frac{\rho_1}{2} \|\nabla_y S - A + \frac{\tau_1}{\rho_1}\|_2^2 \\ &\quad + \frac{\rho_2}{2} \|\nabla_x O - \nabla_x S - B + \frac{\tau_2}{\rho_2}\|_2^2 \\ &\quad + \frac{\rho_3}{2} \|S - C + \frac{\tau_3}{\rho_3}\|_2^2\} \end{aligned} \quad (19)$$

Step 1) Solve

$$A^k = \arg \min_A \|\frac{\rho_1}{2} \|\nabla_y S - A + \frac{\tau_1}{\rho_1}\|_2^2$$

in (16). A^k is solved by iteratively updating A^k by using the following pixel-wise soft-thresholding function [81].

$$A^{k+1} = \text{sign} \left(\frac{\nabla_y S^k + \frac{\tau_1^k}{\rho_1}}{\rho_1} \right) \cdot \max \left(\frac{\nabla_y S^k + \frac{\tau_1^k}{\rho_1}}{\rho_1}, \frac{1}{\rho_1} \right), \quad (20)$$

Step 2) Solve

$$B^k = \arg \min_B \{\|B\|_{wg,1} + \frac{\rho_2}{2} \|\nabla_x O - \nabla_x S - B + \frac{\tau_2}{\rho_2}\|_2^2\}$$

in (17). B^k is solved by iteratively updating B^k by using the following pixel-wise soft-thresholding function and norm weight w_n .

$$B^{k+1} = \text{sign} \left(\frac{\nabla_x O - \nabla_x S^k + \frac{\tau_2^k}{\rho_2}}{\rho_2} \right)$$

$$\max_{\tilde{A}} \|\nabla_x O - \nabla_x S^k + \frac{\tau_2^k}{\rho_2} - \frac{w_n^{k+1} \lambda_1}{\rho_2}, 0\|_2 \quad (21)$$

Step 3) Solve

$$C^k = \arg \min_C \{\|C\|_{wg,2,1} + \frac{\rho_3}{2} \|S - C + \frac{\tau_3}{\rho_3}\|_2^2\}$$

in (18). C^k is solved by iteratively updating C^k by using the following group-wise soft-thresholding function [82] and group norm weight w_g :

$$C_{gi}^{k+1} = \max \left(\eta_{gi}^k - \frac{w_g^{k+1} \lambda_2}{\rho_3}, 0 \right) \cdot \frac{\eta_{gi}^k}{\|\eta_{gi}^k\|_2},$$

$$\eta_{gi}^k = S_{gi}^k + \frac{\tau_3^k}{\rho_3}, \quad (22)$$

where g_i is the group corresponding to the i th column.

Step 4) Solve

$$\begin{aligned} \hat{S}^k &= \arg \min_S \left\{ \frac{\rho_1}{2} \|\nabla_y S - A + \frac{\tau_1}{\rho_1}\|_2^2 \right. \\ &\quad \left. + \frac{\rho_2}{2} \|\nabla_x O - \nabla_x S - B + \frac{\tau_2}{\rho_2}\|_2^2 \right. \\ &\quad \left. + \frac{\rho_3}{2} \|S - C + \frac{\tau_3}{\rho_3}\|_2^2 \right\} \end{aligned}$$

in (19). \hat{S} is a least squares problem, which can be converted into the following linear equation.

$$\begin{aligned} \rho_1 \nabla_y^T \nabla_y \hat{A} \rho_2 \nabla_x^T \nabla_x + \rho_3 \hat{S}^{k+1} \tilde{A} \\ = \rho_1 \nabla_y^T A^{k+1} - \frac{\tau_1^k}{\rho_1} + \rho_2 \nabla_x^T \nabla_x O - B^{k+1} + \frac{\tau_2^k}{\rho_2} \\ + \rho_3 C^{k+1} - \frac{\tau_3^k}{\rho_3} \end{aligned} \quad (23)$$

Then, \hat{S} is solved by iteratively updating S^k by using the following fast Fourier transform (FFT) according to the periodic boundary conditions of S [83]:

$$\begin{aligned} S^{(k+1)} &= F^{-1} \frac{P}{Q}, \\ P &= \rho_1 F(\nabla_y)^* \odot F(A^{k+1} + \frac{\tau_1^k}{\rho_1}) \\ &\quad * \rho_2 F(\nabla_x)^* \odot F(\nabla_x O - B^{k+1} + \frac{\tau_2^k}{\rho_2}) \\ &\quad + \rho_3 F(C^{k+1} - \frac{\tau_3^k}{\rho_3}), \\ Q &= \rho_1 F(\nabla_y)^* \odot F(\nabla_y) + \rho_2 F(\nabla_x)^* \odot F(\nabla_x) + \rho_3 \end{aligned} \quad (24)$$

where F denotes the FFT, and F^{-1} represents the inverse FFT. In addition, $(\cdot)^*$ and \odot represent conjugate transpose and element-wise product respectively.

Step 5) Update Lagrange multipliers.

Lagrange multipliers τ_1^k , τ_2^k and τ_3^k are updated iteratively as follows.

$$\begin{aligned} \square \quad \tau_1^{k+1} &= \tau_1^k + \rho_1 \nabla_y S^{k+1} - A^{k+1}, \\ \tau_2^{k+1} &= \tau_2^k + \rho_2 \nabla_x O - \nabla_x S^{k+1} - B^{k+1}, \\ \blacksquare \quad \tau_3^{k+1} &= \tau_3^k + \rho_3 S^{k+1} - C^{k+1}. \end{aligned} \quad (25)$$

Lagrange multipliers τ_1^k , τ_2^k and τ_3^k support convergence towards the optimal solution of subproblems A , B , C and S .

5) CONVERGENCE CHECK

Our optimization process is terminated if the difference between observed image O and stripe noise component S at successive iterations is within the tolerance as in (26), or if it reaches the maximum iteration number:

$$\frac{\|O - S^{k+1} - O - S^k\|_2}{\|O - S^k\|_2} \leq tol, \quad (26)$$

where tol refers to the tolerance for the stopping criterion, and is set to 10^{-4} considering a balance between convergence speed and accuracy.

C. COMPUTATIONAL COMPLEXITY ANALYSIS

In our ADOM, given an input image $X \in \mathbb{R}^{m \times n}$, where m and n are the number of rows and columns of the image, the computational complexity mainly focuses on solving subproblems A , B , C and S .

- 1) Subproblems A and B : Each iteration takes $O(mn)$, since (20) and (21) mainly include pixel-wise soft-thresholding function.
- 2) Subproblems C : Each iteration takes $O(mn)$, since (22) mainly includes group-wise soft-thresholding function.
- 3) Subproblems S : Each iteration takes $O(mn \log mn)$, since (24) mainly includes FFT.

Adding them up, it takes $O(3mn + mn \log mn)$ at each iteration for all subproblems. Therefore, the total computational complexity of our ADOM can be expressed as $O(k(3mn + mn \log mn))$ for iteration number k .

V. EXPERIMENTAL RESULTS

In this section, we explain various experiments for simulated and real RSI datasets in detail in order to demonstrate the effectiveness of our proposed ADOM. We evaluate the performance of ADOM by comparing it with the following seven distinguished image destriping models with their parameters adjusted for the best performance according to the guide of original literature: wavelet transform and adaptive frequency domain filtering model (WTAF) [20], unidirectional total variation model (UTV) [48], group sparsity-based regularization (GSR) [53], directional L_0 sparse model (DLS) [54], reweighted block sparsity regularization (RBS) [55], low-rank Hodrick-Prescott decomposition model (LRHP) [65] and general destriping framework (GDF) [66]. In the following experiments, the pixel values of RSI are normalized to the range of from 0 to 1. All experiments are performed on

a PC equipped with an Intel Core i5-7600 @ 3.50 GHz and 32-GB of RAM.

A. EXPERIMENTAL DATA

In this section, we shall explain about two different types of experimental image data sets: simulated and real.

1) SIMULATED IMAGE DATA

Simulated image data are useful for comparing different stripe noise removal models, since it can be generated by adding various types of strip noises with different intensities into the reference image. Typically, the reference image is referred as the one without observed stripe patterns. We use Cuprite and Pavia University data sets as the reference image. The former is acquired by an Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor for exploring underground mineral resources. Each image has a size of $614 \times 512 \times 224$. For our experiment, we use a 400×400 sub-image with the 10th band. The latter is collected by a Reflective Optics System Imaging Spectrometer (RODIS) sensor for analyzing the Earth's surface. Each image has a size of $610 \times 340 \times 103$. For our experiment, we use a 300×300 sub-image with the 103rd band.

We generate simulated image data sets by adding five different types of stripe noise into Cuprite and Pavia University datasets as follows:

Case 1) Periodical Stripe: Periodical stripes with the same noise intensity between -40 and 40 are added to the reference image with 40% noise ratio with respect to the image width.

Case 2) Non-Periodical Stripe: Non-periodical stripes with different noise intensities between -40 and 40 are added to the reference image with 40% noise ratio with respect to the image width.

Case 3) Broken Stripe: Broken stripes with different noise intensities between -40 and 40 are added to the reference image with 20% noise ratio with respect to the image width, with each stripe randomly selected in length from the image height.

Case 4) Multiplicative Stripe: Multiplicative stripes with gain ranging from 0.8 to 1.2 and different offsets between -40 and 40 are added to the reference image with 60% noise ratio with respect to the image width.

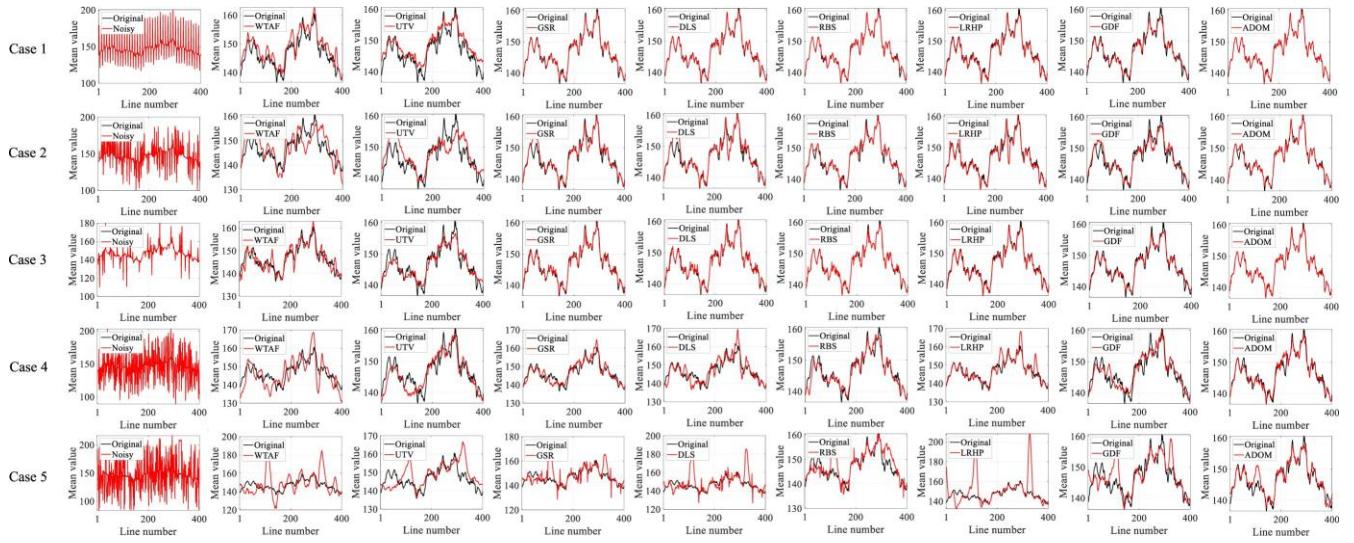
Case 5) Non-Periodical + Broken + Wide Stripe: To provide a more comprehensive evaluation, non-periodical, broken and wide stripes with noise intensities ranging from -60 to 60 are added to the reference image. The noise ratios for non-periodical and broken stripes are 25% and 70% respectively for the image width, while the noise ratio for the wide stripe is 5%.

2) REAL IMAGE DATA

Real image data are used to compare our ADOM with the other image destriping models in the real environment with stipe noise. We use four real image data sets: Hyperion, Moon Mineralogy Mapper (M3), Aqua Moderate Resolution Imaging Spectroradiometer (Aqua MODIS)

TABLE 1. Quantitative comparison of various models for simulated image data.

Simulated Image Data	Metric	Noisy	WTAF	UTV	GSR	DLS	RBS	LRHP	GDF	ADOM	
Cuprite	Case 1	PSNR	25.1058	39.1555	39.3682	50.4970	52.9976	53.1205	53.3515	38.9048	54.9151
		SSIM	0.4023	0.9859	0.9856	0.9938	0.9962	0.9971	0.9990	0.9373	0.9978
		D	6.0222	2.2641	2.2172	0.5597	0.3494	0.2301	0.4506	1.9904	0.1978
	Case 2	PSNR	24.8744	34.3737	36.9042	43.8306	41.4955	48.1677	45.0591	37.7040	50.4106
		SSIM	0.4418	0.9819	0.9847	0.9922	0.9907	0.9953	0.9950	0.9344	0.9958
		D	7.9494	4.0821	2.9182	1.0223	1.1076	0.4976	0.7436	2.3680	0.4246
	Case 3	PSNR	30.8206	34.3540	40.9965	39.6778	39.6604	40.9431	35.9966	38.9707	49.7776
		SSIM	0.8031	0.8714	0.9802	0.9544	0.9527	0.9653	0.8912	0.9352	0.9933
		D	2.0084	2.9051	1.6655	1.1419	0.9431	0.6882	1.5855	1.9594	0.2866
	Case 4	PSNR	21.2988	33.4643	38.3192	40.0207	37.2623	42.0291	37.6530	36.2996	43.1783
		SSIM	0.2514	0.9593	0.9775	0.9820	0.9752	0.9843	0.9706	0.9315	0.9851
		D	16.4393	4.4875	2.4429	1.9004	2.4913	1.3458	1.9513	2.9423	1.1475
	Case 5	PSNR	18.3992	25.6277	32.3761	29.6189	26.8228	34.4926	24.0628	33.9028	41.5778
		SSIM	0.1982	0.7516	0.9712	0.8544	0.8288	0.9406	0.7445	0.9177	0.9828
		D	21.9156	9.5793	4.6464	4.6165	6.1627	2.7899	8.2888	3.6642	1.3895
Pavia University	Case 1	PSNR	25.1242	37.5511	39.3080	51.8180	54.4421	53.7940	45.3563	42.4683	55.7431
		SSIM	0.6134	0.9590	0.9857	0.9948	0.9954	0.9960	0.9878	0.9794	0.9967
		D	7.7053	3.6142	2.0482	0.5605	0.4740	0.3240	1.1565	1.4938	0.3046
	Case 2	PSNR	25.2198	35.0523	39.2446	50.0254	50.6893	51.8887	44.2812	41.6129	53.3419
		SSIM	0.8498	0.8924	0.9816	0.9465	0.9416	0.9609	0.9168	0.9725	0.9936
		D	2.0807	3.3744	1.6088	1.2601	1.2174	0.7737	1.9767	1.4618	0.3079
	Case 3	PSNR	30.6747	33.9845	40.7232	37.5213	37.0138	38.8779	36.2047	40.5760	47.0337
		SSIM	0.8427	0.8911	0.9173	0.9832	0.9177	0.9397	0.9368	0.9588	0.9959
		D	2.0138	3.2382	2.222	1.5168	2.3781	1.3294	1.2550	0.7140	0.2301
	Case 4	PSNR	22.8615	32.3472	36.3329	37.4059	36.0269	37.1282	36.0701	36.2006	38.3713
		SSIM	0.4958	0.9350	0.9749	0.9745	0.9673	0.9764	0.9575	0.9685	0.9771
		D	13.2852	4.9614	2.8466	2.3154	2.6857	2.2474	2.7349	2.8076	1.9144
	Case 5	PSNR	18.0287	24.3516	32.0970	27.6921	25.0586	32.1616	24.0623	32.9548	40.1915
		SSIM	0.3341	0.7503	0.9682	0.8189	0.7918	0.9222	0.7541	0.9423	0.9818
		D	23.0079	11.4221	4.8099	5.6557	7.6003	2.8956	8.7966	3.7003	1.4089

**FIGURE 4.** Mean vertical profiles for Cuprite.

and Terra Moderate Resolution Imaging Spectroradiometer (Terra MODIS). The Hyperion is collected by the Hyperion spectrometer sensor for Earth observation. Each image has a size of $256 \times 3242 \times 242$. For our experiments, we use a 256×256 sub-image with the 93rd band. The M3 is

collected by the near-infrared imaging spectrometer sensor in order to investigate the mineralogical composition and geological characteristics of the Moon. Each image has a size of $304 \times 12036 \times 85$. For our experiments, we use a 300×300 sub-image with the 10th band. The Aqua

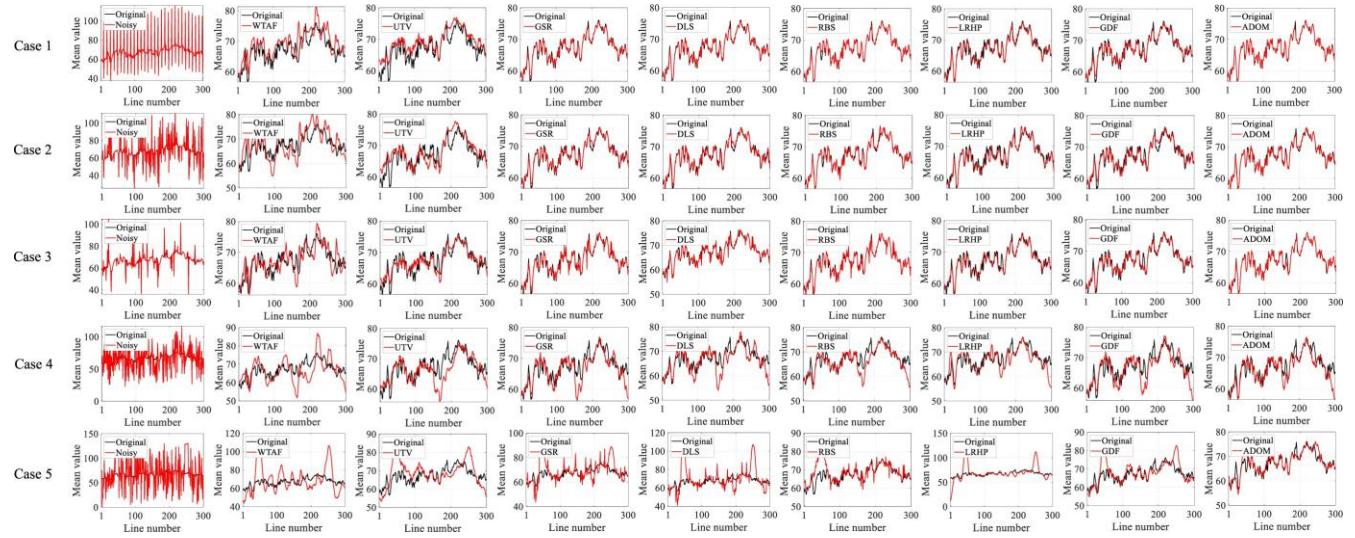


FIGURE 5. Mean vertical profiles for Pavia University.

TABLE 2. Quantitative comparison of various models for real image data.

Real Image Data	Metric	Noisy	WTAF	UTV	GSR	DLS	RBS	LRHP	GDF	ADOM
Hyperion	ICV	28.95	41.27	40.94	55.75	53.87	45.95	54.91	55.95	59.26
	MRD	-	3.36%	4.37%	2.77%	2.65%	2.91%	2.75%	3.03%	2.57%
M3	ICV	30.07	52.62	54.59	50.33	50.26	53.85	49.43	54.41	55.70
	MRD	-	2.17%	2.80%	1.84%	1.63%	1.69%	1.72%	2.17%	1.56%
Aqua MODIS	ICV	84.54	226.99	208.16	217.77	237.54	185.07	212.96	275.21	286.83
	MRD	-	1.02%	1.31%	0.94%	0.86%	0.87%	0.95%	1.12%	0.84%
Terra MODIS	ICV	90.20	160.86	161.06	225.25	235.87	208.10	178.41	182.44	256.61
	MRD	-	1.54%	1.44%	1.08%	1.19%	1.41%	1.50%	1.16%	1.01%

MODIS is collected by the MODIS sensor in order to gather information on the Earth's water cycle. Each image has a size of $1354 \times 2030 \times 36$. For our experiments, we use a 1000×1000 sub-image with the 30th band. The Terra MODIS is acquired by the MODIS sensor in order to observe changes in the Earth's surface. Each image has a size of $1354 \times 2030 \times 36$. For our experiments, we use a 1000×1000 sub-image with the 25th band.

B. QUANTITATIVE EVALUATION

We make use of five evaluation metrics for each of simulated and real image data: peak signal-to-noise ratio (PSNR), structural similarity index measure (SSIM), degree of distortion (D), inverse coefficient of variation (ICV) and mean relative deviation (MRD). PSNR measures the ratio between the maximum possible power of the image signal and the power of the noise signal as shown below:

$$\text{PSNR} = 10\log_{10} \frac{\sum_{i=1}^m \sum_{j=1}^n [I(i,j) - \hat{I}(i,j)]^2}{\sum_{i=1}^m \sum_{j=1}^n I(i,j)^2}, \quad (27)$$

where I is the original image, \hat{I} is the noisy image and MAX_I is the maximum possible pixel value of the original image.

SSIM measures the degree of structural changes between two images as shown below:

$$\text{SSIM} = \frac{(2\mu_I \hat{\mu}_I + c_1)(2\sigma_I \hat{\sigma}_I + c_2)}{(\mu_I^2 + \hat{\mu}_I^2 + c_1)(\sigma_I^2 + \hat{\sigma}_I^2 + c_2)}, \quad (28)$$

where μ_I , $\hat{\mu}_I$, σ_I and $\hat{\sigma}_I$ are the mean and standard deviation of I and \hat{I} respectively. Additionally, $\sigma_I \hat{\sigma}_I$ is the covariance between \hat{I} and I , and c_1 and c_2 are constants used to prevent division by zero error.

D measures the average absolute difference between the original image and the noisy image as shown below.

$$D = \frac{\frac{P_m}{i=1} \frac{P_n}{j=1} |I(i,j) - \hat{I}(i,j)|}{m \times n} \quad (29)$$

As shown below, ICV measures the ratio of mean with respect to standard deviation for the selected region where changes between image pixels are small:

$$\text{ICV} = \frac{R_m}{R_s}, \quad (30)$$

where R_m and R_s are the mean and standard deviation of the selected region respectively.

MRD measures the distortion between destriped and input images for a selected region with large changes in pixel values

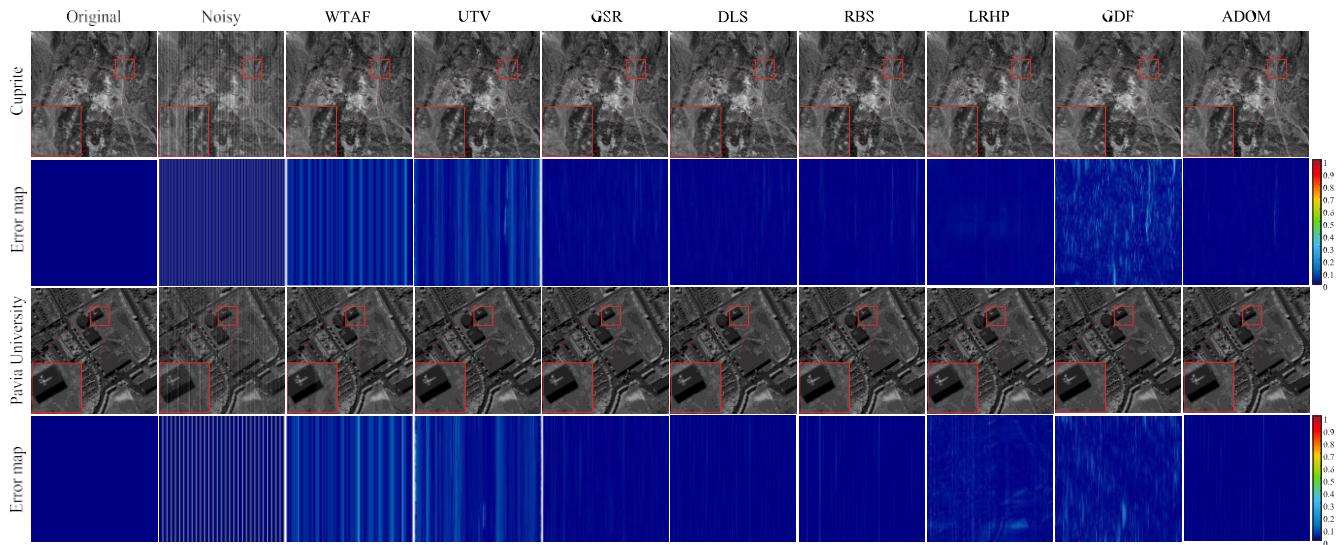


FIGURE 6. Destriping results and its error maps on simulated image data for case 1 of Cuprite and Pavia University.

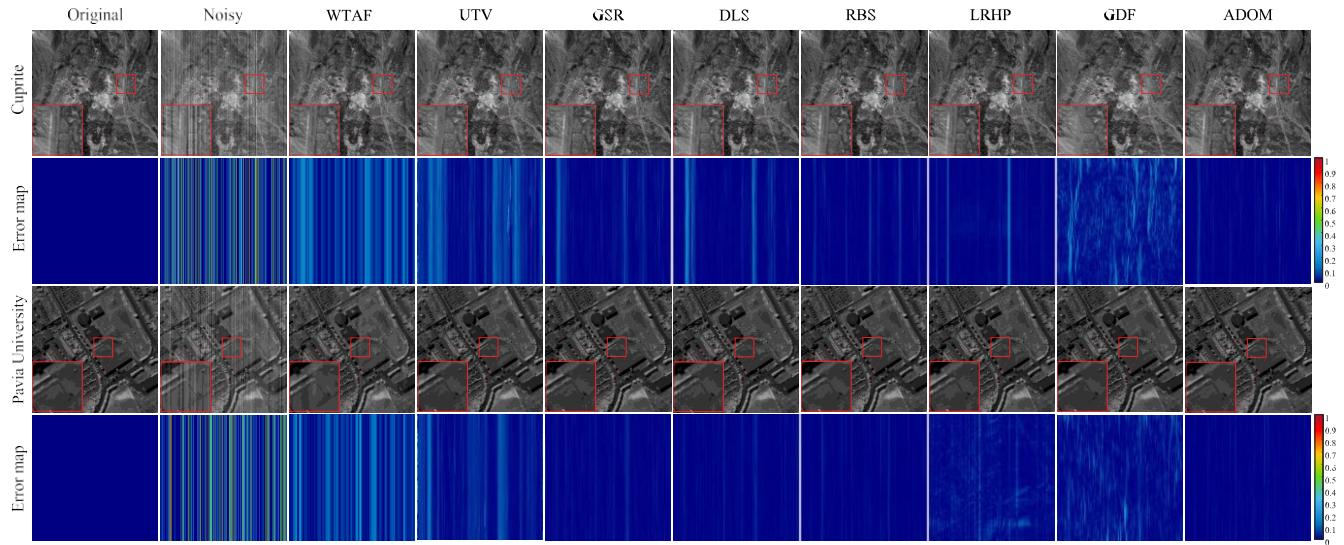


FIGURE 7. Destriping results and its error maps on simulated image data for case 2 of Cuprite and Pavia University.

as shown below:

$$\text{MRD} = \frac{1}{m \times n} \sum_{i=1}^m \sum_{j=1}^n \frac{|R_d(i, j) - R_i(i, j)|}{R_i(i, j)} \times 100, \quad (31)$$

where R_d and R_i are the destriped and input images of selected region respectively. For the experiments, we select two distinct 10×10 regions for ICV and MRD respectively. In PSNR, SSIM and ICV, the higher values indicate better results in stripe noise removal, whereas the lower values of D and MRD indicate better results.

1) SIMULATED IMAGE DATA

Table 1 lists PSNR, SSIM and D results of various image destriping models for Cuprite and Pavia University under five different noise cases, where the best value for each quality metric is highlighted in bold. As can be seen from Table 1,

our ADOM shows better performance than all the other image destriping models in all the cases. Specifically, ADOM demonstrates significant improvement in case 5 for Cuprite and case 3 for Pavia University respectively. Fig. 4 shows the mean vertical profile for all the cases of Cuprite. In the mean vertical profile, the average values in each column of the image are shown. Our ADOM demonstrates more similarity to original image than all the other image destriping models for all cases. Fig. 5 shows the mean vertical profile for all cases of Pavia University. Our ADOM shows more similarity to original image than all the other image destriping models for all cases.

2) REAL IMAGE DATA

Table 2 lists ICV and MRD results for image destriping models on real image data, where the best value for each

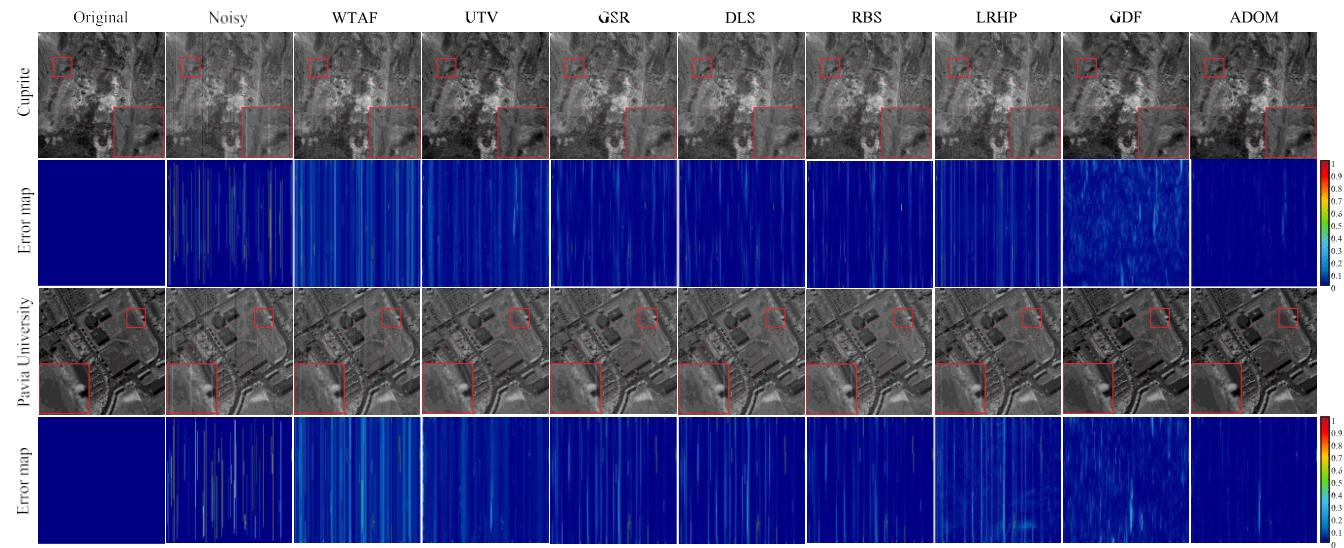


FIGURE 8. Destriping results and its error maps on simulated image data for case 3 of Cuprite and Pavia University.

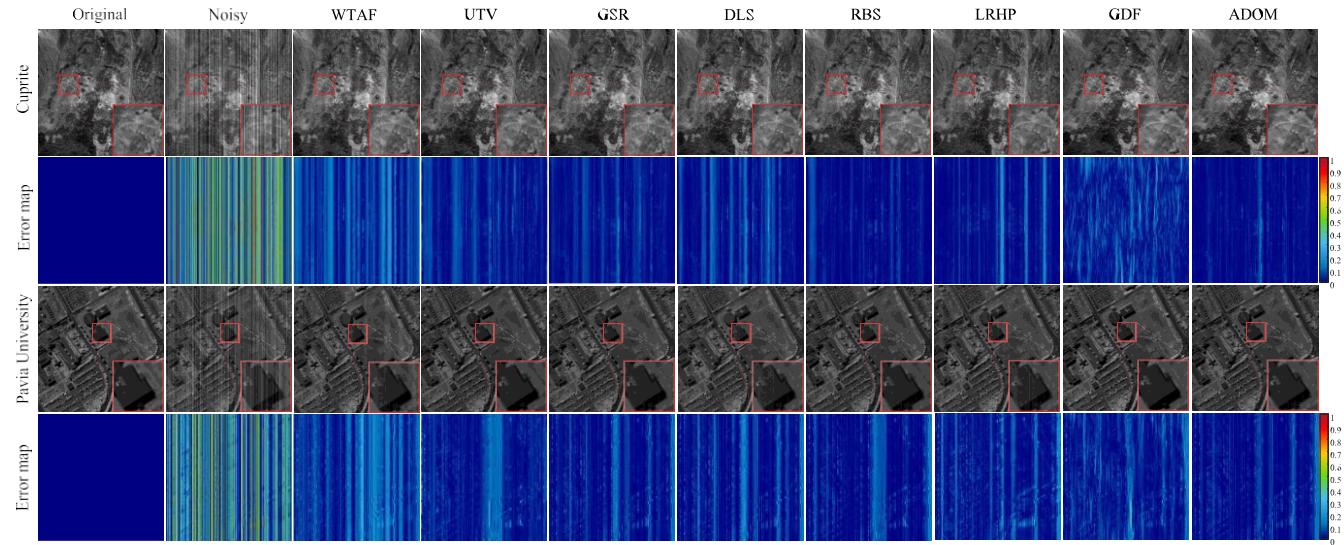


FIGURE 9. Destriping results and its error maps on simulated image data for case 4 of Cuprite and Pavia University.

quality metric is highlighted in bold. As shown in Table 2, ADOM demonstrates the better performance compared to all the other models in real image data with the highest ICV and lowest.

C. QUANTITATIVE EVALUATION

In this section, we shall explain about the qualitative evaluation for each of simulated and real image data.

1) SIMULATED IMAGE DATA

We shall show the destriping results and its error maps for simulated image data of Cuprite and Pavia University in Fig. 6-10. Each error map visualizes the destriping result by displaying the normalized image obtained by eliminating the stripe noises from the original image. As shown in Fig. 6

for case 1, ADOM and LRHP obtain satisfactory visual results while better preserving the image details compared to other models. In addition, the other methods UTV, GSR, DLS, RBS and GDF show quite satisfactory visual results. In contrast, WTAFF may remove stripe noise, but it shows unsatisfactory visual results due to some artifacts. As shown in Fig. 7 for case 2, ADOM shows the more enhanced visual results, which preserves the image details compared to other models. The visual result of WTAFF contains some artifacts, whereas the other models UTV, GSR, DLS, RBS, LRHP and GDF show the loss of the image details. Similarly, as shown in Fig. 8 for case 3, ADOM shows the better visual results, which preserve the image details compared to other models. In contrast, UTV and GDF may remove stripe noise, but it shows more loss of the image details than ADOM. The other models WTAFF, GSR, DLS, RBS

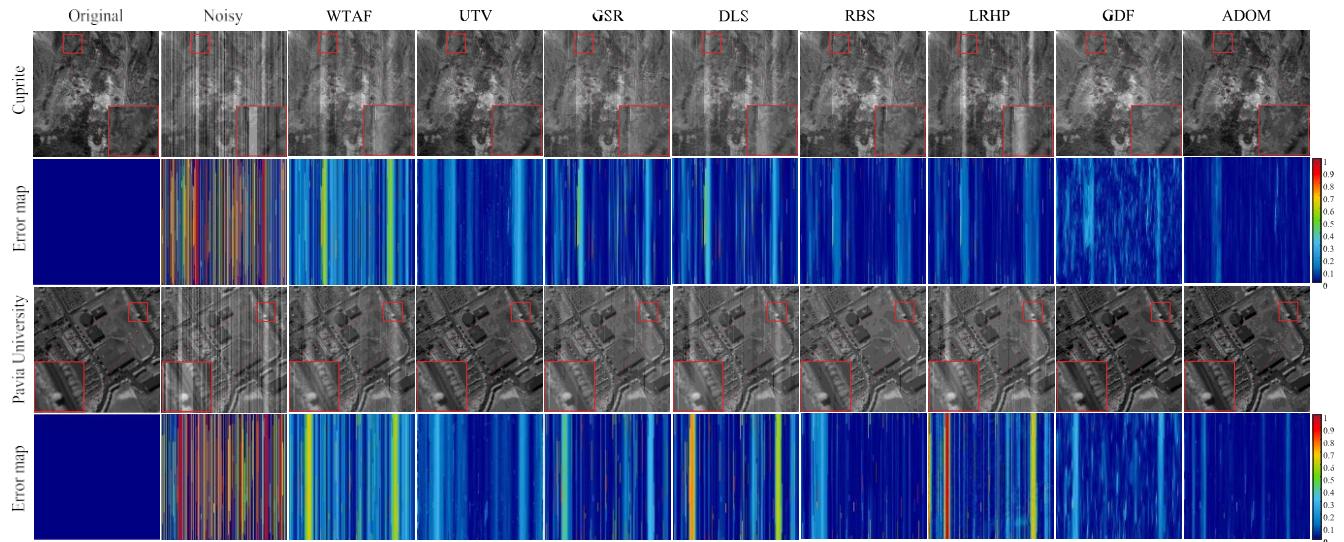


FIGURE 10. Destriping results and its error maps on simulated image data for case 5 of Cuprite and Pavia University.

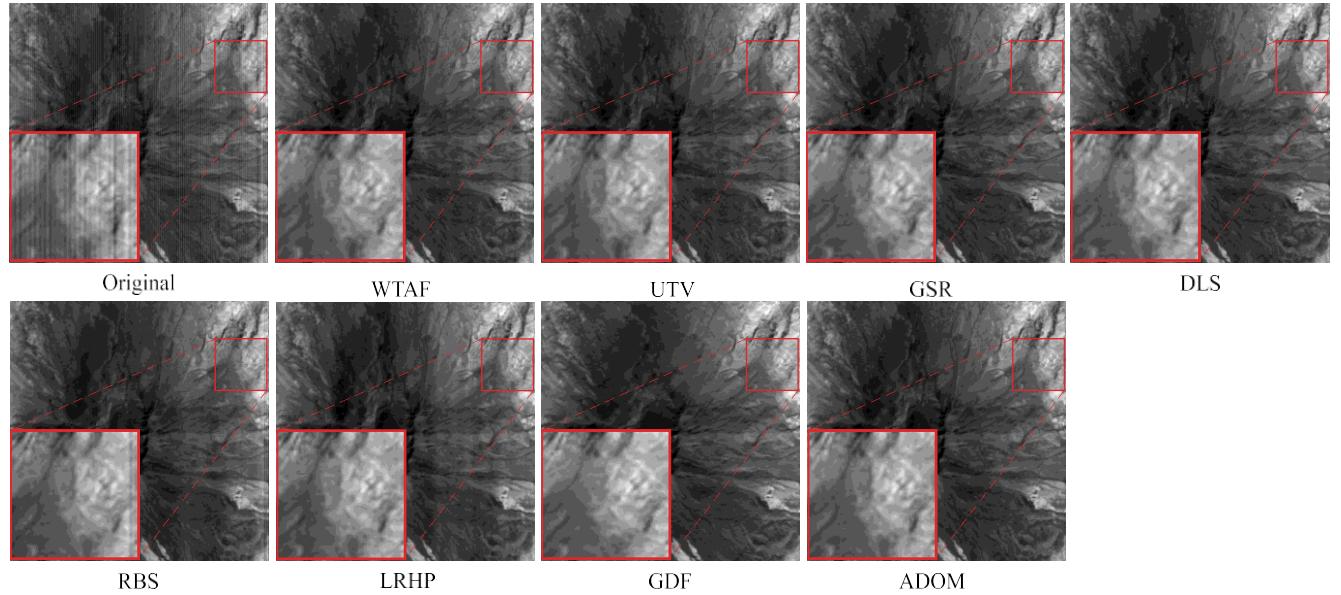


FIGURE 11. Qualitative comparison on real image data for Hyperion.

and LRHP show unsatisfactory visual results due to some unremoved stripe noises compared to ADOM. As shown in Fig. 9 for case 4, ADOM shows better visual results than other models, which may remove the stripe noise, but fail to preserve the image details compared to ADOM. As shown in Fig. 10 for case 5, ADOM shows the better visual results by effectively removing the stripe noise compared to the other models. On the other hand, UTV and LRHP may remove the stripe noise, but it fails to preserve the image details compared to ADOM. The other models WTA, GSR, DLS, RBS and GDF show unsatisfactory visual results compared to ADOM. For all cases, our ADOM achieves better visual improvement for simulated image data than the other image destriping models.

2) REAL IMAGE DATA

We shall show the destriping results for each of Hyperion, M3, Aqua MODIS and Terra MODIS respectively in Fig. 11-14. As shown in Fig. 11 and 12, ADOM shows the better visual results, which preserve the image details compared to the other models. In addition, the other models, UTV, GSR and DLS demonstrate the visual results similar to ADOM. However, WTA and GDF demonstrate unsatisfactory visual results due to some artifacts and excessive blurring effects respectively. RBS has some stripe noises unremoved. As shown in Fig. 13 and 14, ADOM exhibits the better visual results, which preserve the image details compared to the other models. UTV, GSR, DLS and RBS show the visual results similar to ADOM. However, WTA and LRHP

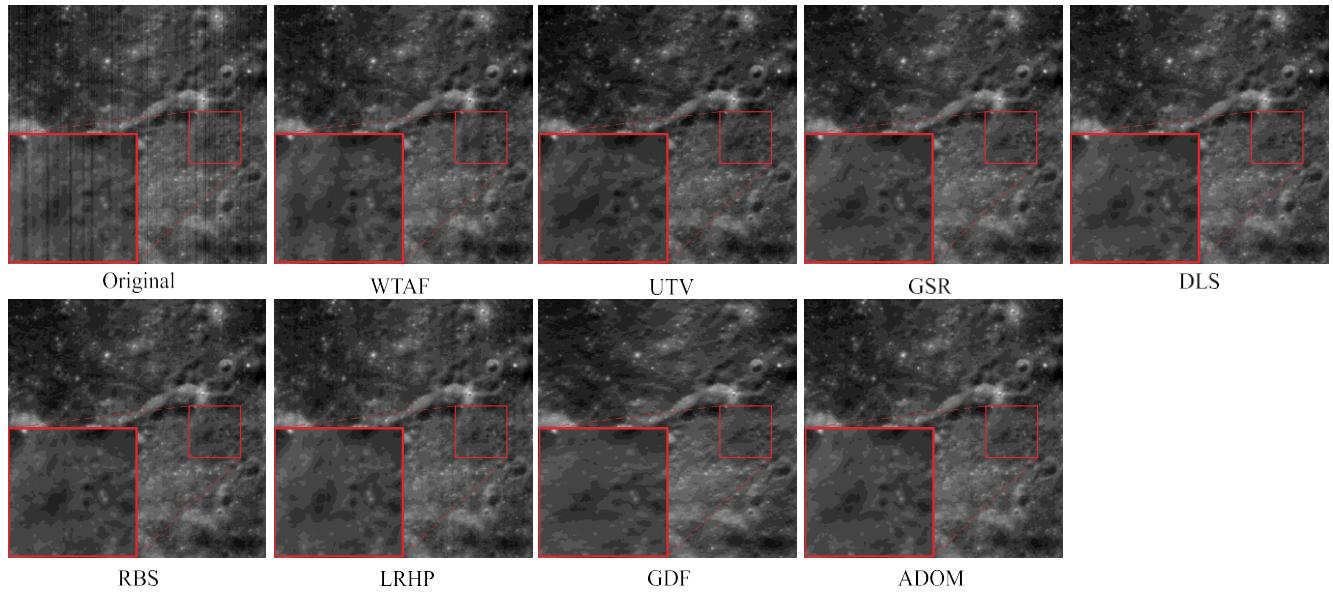


FIGURE 12. Qualitative comparison on real image data for M3.

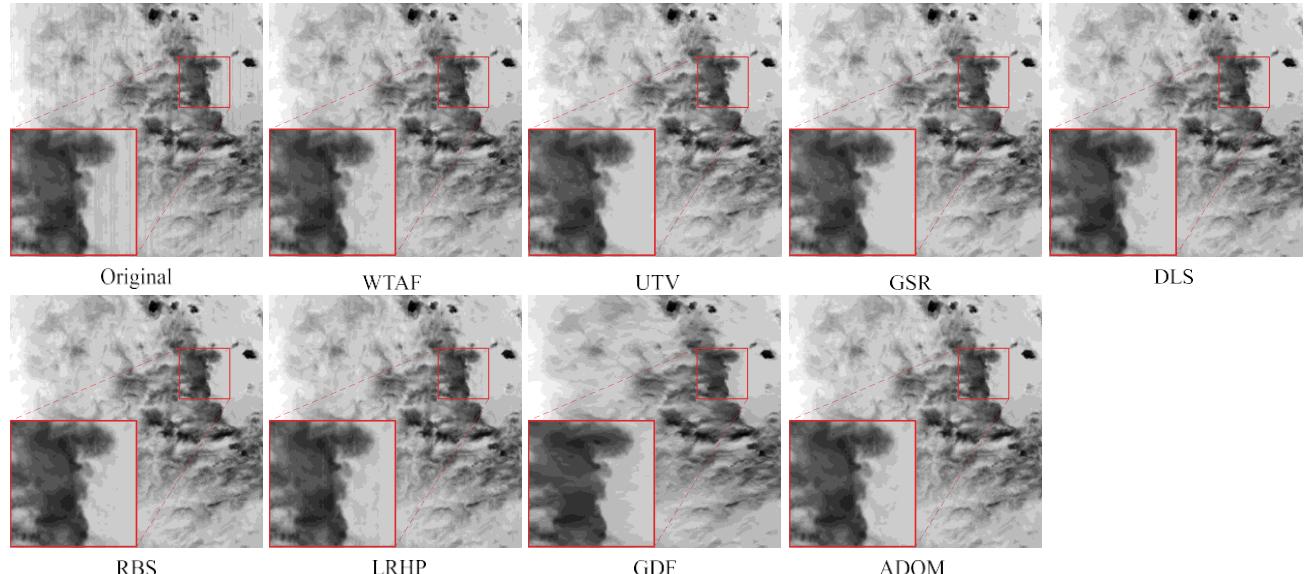


FIGURE 13. Qualitative comparison on real image data for Aqua MODIS.

show unsatisfactory visual results due to some artifacts. GDF shows undesirable visual results due to excessive blurring effects.

D. EMPIRICAL ANALYSIS

In this section, we shall describe the results of our empirical analysis in order to gain further insight into the performance of ADOM under different conditions. These results include noise intensity, iteration number, regularization parameters selection, threshold parameter selection, ablation study and running time.

1) NOISE INTENSITY

We shall compare the performance between various models for noise intensity. To this end, we consider various levels of noise intensity for all cases of Cuprite. Fig. 15 shows PSNR and SSIM with respect to noise intensity of Cuprite. Our ADOM demonstrates superior performance in most cases, although it obtains unsatisfactory results in some cases. For case 1, SSIM of ADOM is lower than that of LRHP for all noise intensities. However, PSNR of ADOM is better than that of LRHP for all noise intensities. For case 2, SSIM of ADOM is lower than that of LRHP for the noise intensities 10. However, PSNR and SSIM of ADOM are

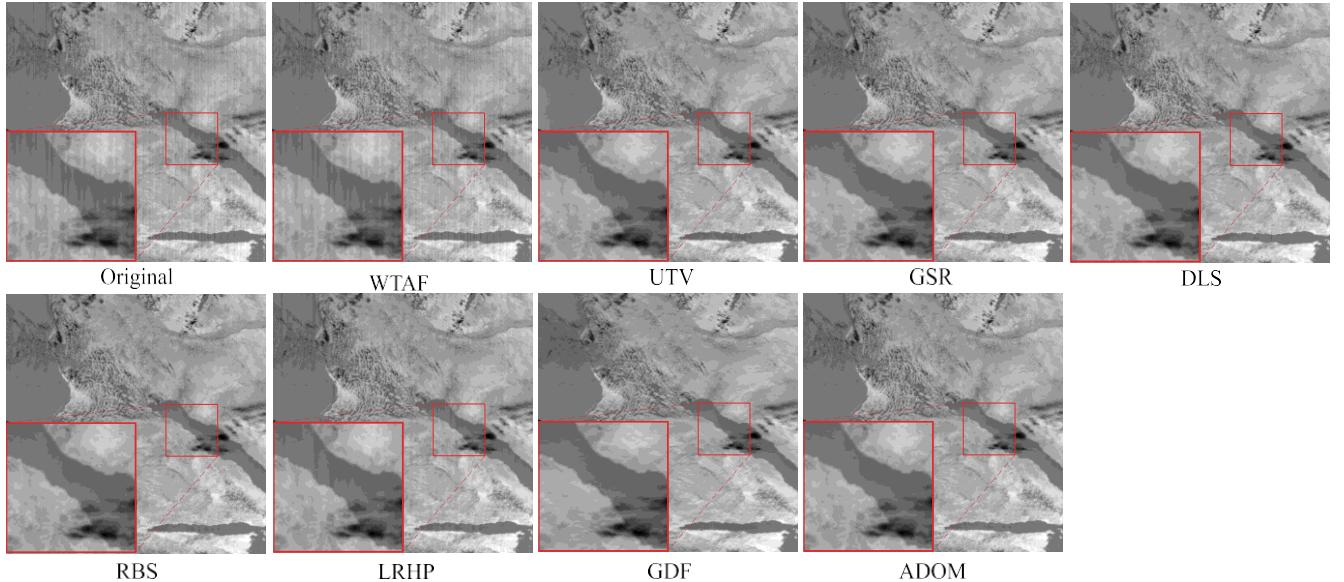


FIGURE 14. Qualitative comparison on real image data for Terra MODIS.

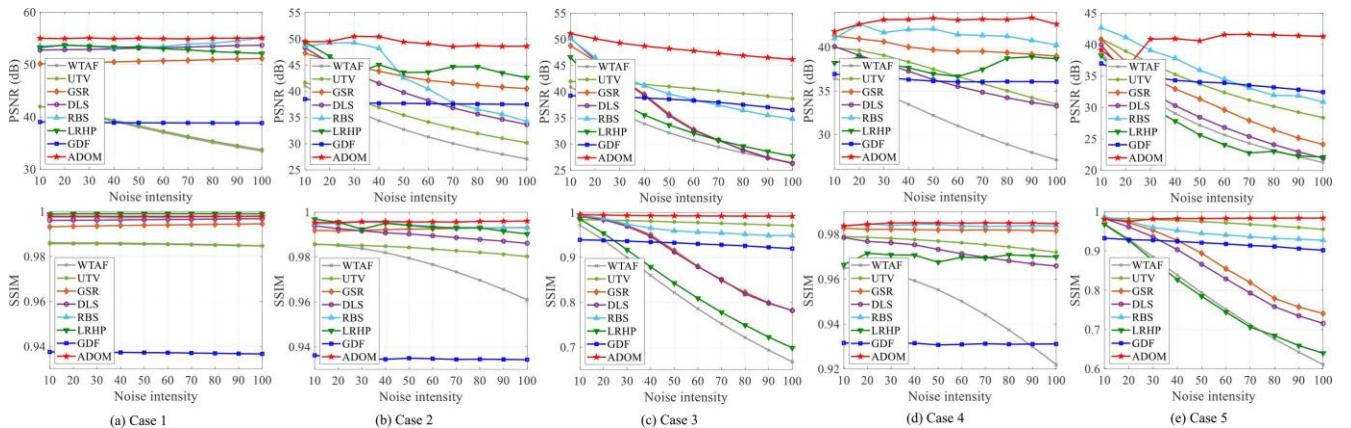


FIGURE 15. PSNR and SSIM w.r.t noise intensity.

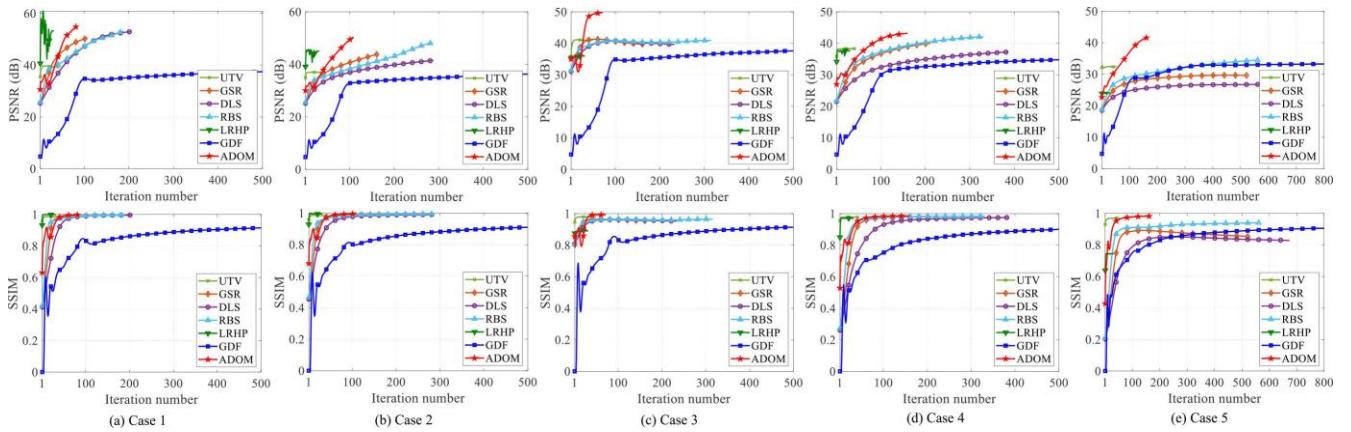


FIGURE 16. PSNR and SSIM w.r.t iteration number.

better than those of LRHP for noise intensities above 20. For cases from 3 to 4, ADOM shows better performance than other models. For case 5, ADOM demonstrates a lower

performance than the other models for noise intensities below 20. However, ADOM shows much better performance than the other models for noise intensities ranging from 30 to 100.

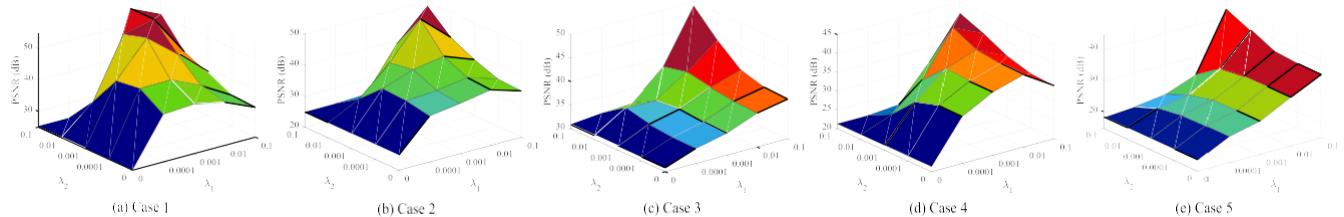


FIGURE 17. PSNR w.r.t λ_1 and λ_2 .

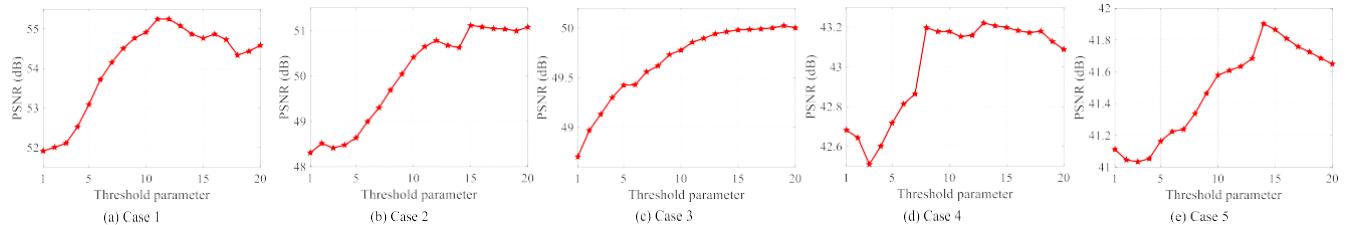


FIGURE 18. PSNR w.r.t threshold parameter.

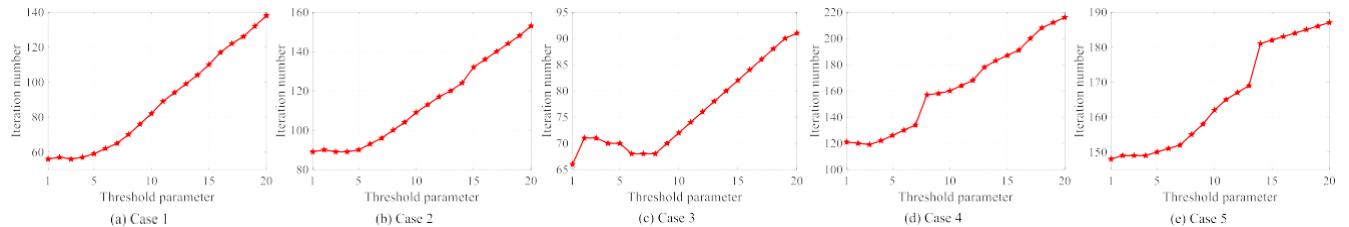


FIGURE 19. Iteration number w.r.t threshold parameter.

In summary, ADOM demonstrates relatively better performance compared to other models as the noise intensity increases, which shows that ADOM is particularly effective in handling higher levels of noise.

2) ITERATION NUMBER

In this section, we shall compare the performance between various optimization-based models for iteration number. Fig. 16 shows PSNR and SSIM with respect to iteration number of Cuprite. ADOM achieves a higher PSNR and SSIM than other models for all cases. In particular, ADOM demonstrates better performance with fewer iteration number.

3) REGULARIZATION PARAMETERS SELECTION

We compare the performance of ADOM for various λ_1 and λ_2 of Cuprite as shown in Fig. 17. For cases 1, 2 and 4, ADOM with λ_1 and λ_2 set to 0.01 and 0.01 respectively demonstrates better performance than the other λ_1 and λ_2 . On the other hand, for cases 3 and 5, ADOM with λ_1 and λ_2 set to 0.1 and 0.1 respectively shows better performance than the other λ_1 and λ_2 . Based on these results, λ_1 and λ_2 are empirically set to ($\lambda_1 = 0.01, \lambda_2 = 0.01$) or ($\lambda_1 = 0.1, \lambda_2 = 0.1$) depending on the type of stripe noise in all experiments.

4) THRESHOLD PARAMETER SELECTION

In this section, we shall compare PSNR and iteration number of ADOM for various threshold parameters of Cuprite respectively. For all cases, ADOM demonstrates a stable PSNR for around the threshold parameter to 10 as shown in Fig. 18. Fig. 19 shows the iteration number with respect to threshold parameter. For all cases, ADOM shows a continuous increase in the iteration number as the threshold parameter increases. Our experimental results show that ADOM achieves the best PSNR performance when threshold parameter p is set to 10 compared to other threshold parameters.

5) ABLATION STUDY

In this section, we compare the performance of ADOM for 8 cases of weight control, starting point control and step-size control. Table 3 lists the ablation study on simulated image data for case 2 of Cuprite. In Table 3, the best results for each quality metric are highlighted in bold. Case 2 using only weight control has the better performance than case 1 using no control strategies for all metrics.

Case 3 using only starting point control has the poor performance compared to case 2 using only weight control for all metrics. Case 4 using only step-size control has the better performance than case 2 using only weight control

TABLE 3. Ablation study on simulated image data for case 2 of Cuprite.

Case	Strategy			Metric			
	Weight Control	Starting Point Control	Step-Size Control	PSNR	SSIM	D	Iteration Number
Case 1				43.8306	0.9922	1.0223	162
Case 2	✓			48.9368	0.9954	0.4806	159
Case 3		✓		43.8306	0.9922	1.0223	162
Case 4			✓	43.8907	0.9901	1.0719	98
Case 5	✓	✓		48.9395	0.9954	0.4794	159
Case 6	✓		✓	48.4315	0.9938	0.5562	92
Case 7		✓	✓	44.1492	0.9907	1.0268	99
Case 8	✓	✓	✓	50.4106	0.9958	0.4246	109

TABLE 4. Running time.

Image	WTAF	UTV	GSR	DLS	RBS	LRHP	GDF	ADOM
Hyperion	0.04	0.33	0.59	1.36	2.39	0.33	5.65	0.52
M3	0.07	0.49	0.76	1.52	3.61	0.45	5.09	0.60
Aqua MODIS	0.34	3.86	6.99	13.91	74.1	5.06	11.12	6.12
Terra MODIS	0.27	4.22	8.18	17.14	52.53	4.69	16.99	7.30

for iteration number. Case 5 using weight and starting point control has the better performance than the other cases with only one control strategy for PSNR and D. Case 6 using weight and step-size control has the better performance than the other cases for iteration number. Case 7 using starting point and step-size control has the better performance than case 4 for PSNR, SSIM and D. Finally, case 8 using three control strategies shows the best performance compared to the other cases for PSNR, SSIM and D. In summary, our results demonstrate that not only can ADOM derive the best performance from using all strategies, but also that some strategies can yield performance improvements.

6) RUNNING TIME

In this section, we shall compare the running time between various models. Table 4 lists the running time of various models for different image data sets: Hyperion, M3, Aqua MODIS and Terra MODIS. For all data sets, even though WTAF, UTV and LRHP have better performance than ADOM in terms of running time. ADOM has a better performance than those in terms of ICV and MRD as shown in Table 2. For all data sets, ADOM has a better performance than GSR, DLS, RBS and GDF not only in terms of running time, but also ICV and MRD as shown in Table 2.

VI. CONCLUSION

In this paper, we have presented a new ADMM-based optimization model, called ADOM for stripe noise removal in RSI. We have formulated the objective function for ADOM. After converting it into constrained objective function, we have developed augmented Lagrangian function which shall be used to find stripe noise components. We have described about optimization process for solving the optimization function in order to efficiently extract stripe noise component. While the existing optimization-based

models mainly focus on enhancing visual quality, ADOM not only employs a weight-based detection strategy for efficient stripe noise component capture, but also ADMM-based acceleration strategy for fast stripe noise removal. In the weight-based detection strategy, we effectively detect stripe noise similar to the image details by using weighted norm generated by adjusting norm and group norm weights based on the momentum coefficient and residual parameter. In the ADMM-based acceleration strategy, we accelerate optimization process by using two control strategies: evidence-based starting point control and momentum-based step-size control. Evidence-based starting point control updates momentum coefficient and damping coefficient by properly adjusting the threshold parameter, and hence provides a starting point for more accurately finding stripe noise component. Momentum-based step-size control accelerates convergence by using the momentum coefficient while providing optimization stability by exploiting the damping coefficient. The performance of ADOM is demonstrated through various experiments in terms of quantitative and qualitative evaluations. We have shown that ADOM achieves better performance than other despeckling models for each of simulated and real image data sets.

In the future, we shall plan to expand our research into image dehazing, which is emerging as an important issue in the field of image processing, and continue to develop an efficient deep learning model for image dehazing.

REFERENCES

- [1] S. Shi, Y. Zhong, J. Zhao, P. Lv, Y. Liu, and L. Zhang, “Land-use/land-cover change detection based on class-prior object-oriented conditional random field framework for high spatial resolution remote sensing imagery,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5600116.
- [2] J. Hemmerling, D. Pflugmacher, and P. Hostert, “Mapping temperate forest tree species using dense Sentinel-2 time series,” *Remote Sens. Environ.*, vol. 267, Dec. 2021, Art. no. 112743.
- [3] B. Zhang, D. Wu, L. Zhang, Q. Jiao, and Q. Li, “Application of hyperspectral remote sensing for environment monitoring in mining areas,” *Environ. Earth Sci.*, vol. 65, no. 3, pp. 649–658, Feb. 2012.
- [4] X. Zheng, J. Ye, Y. Chen, S. Wistar, J. Li, J. A. P. Fernández, M. A. Steinberg, and J. Z. Wang, “Detecting comma-shaped clouds for severe weather forecasting using shape and motion,” *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 6, pp. 3788–3801, Jun. 2019.
- [5] M. Abrams, D. Pieri, V. Realmutto, and R. Wright, “Using EO-1 hyperion data as HypsIRI preparatory data sets for volcanology applied to Mt Etna, Italy,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 2, pp. 375–385, Apr. 2013.
- [6] H. Wang, H. Cao, Y. Kai, H. Bai, X. Chen, Y. Yang, L. Xing, and C. Zhou, “Multi-source remote sensing intelligent characterization technique-based disaster regions detection in high-altitude mountain forest areas,” *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 3512905.
- [7] G. Chander, T. J. Hewison, N. Fox, X. Wu, X. Xiong, and W. J. Blackwell, “Overview of intercalibration of satellite instruments,” *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 3, pp. 1056–1080, Mar. 2013.
- [8] J. Chen, Y. Shao, H. Guo, W. Wang, and B. Zhu, “Destriping CMODIS data by power filtering,” *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 9, pp. 2119–2124, Sep. 2003.
- [9] H. Carfantan and J. Idier, “Statistical linear destriping of satellite-based pushbroom-type images,” *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 4, pp. 1860–1871, Apr. 2010.
- [10] J. Han, D. Zhang, G. Cheng, L. Guo, and J. Ren, “Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning,” *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 6, pp. 3325–3337, Jun. 2015.

- [11] G. Cheng, P. Zhou, and J. Han, "Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7405–7415, Dec. 2016.
- [12] J. Han, J. Ding, J. Li, and G.-S. Xia, "Align deep features for oriented object detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5602511.
- [13] W. Li, G. Wu, F. Zhang, and Q. Du, "Hyperspectral image classification using deep pixel-pair features," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 844–853, Feb. 2017.
- [14] Z. Zhong, J. Li, Z. Luo, and M. Chapman, "Spectral–spatial residual network for hyperspectral image classification: A 3-D deep learning framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 847–858, Feb. 2018.
- [15] L. Sun, G. Zhao, Y. Zheng, and Z. Wu, "Spectral–spatial feature tokenization transformer for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5522214.
- [16] J. J. Simpson, J. R. Stitt, and D. M. Leath, "Improved finite impulse response filters for enhanced destriping of geostationary satellite data," *Remote Sens. Environ.*, vol. 66, no. 3, pp. 235–249, Dec. 1998.
- [17] J. Guo Liu and G. Llewellyn Keith Morgan, "FFT selective and adaptive filtering for removal of systematic noise in ETM+ imageodesy images," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 12, pp. 3716–3724, Dec. 2006.
- [18] J. Torres and S. O. Infante, "Wavelet analysis for the elimination of striping noise in satellite images," *Opt. Eng.*, vol. 40, no. 7, pp. 1309–1314, Jul. 2001.
- [19] J. Chen, H. Lin, Y. Shao, and L. Yang, "Oblique striping removal in remote sensing imagery based on wavelet transform," *Int. J. Remote Sens.*, vol. 27, no. 8, pp. 1717–1723, Feb. 2007.
- [20] B. Münch, P. Trtik, F. Marone, and M. Stampanoni, "Stripe and ring artifact removal with combined wavelet-Fourier filtering," *Optics Exp.*, vol. 17, no. 10, pp. 8567–8591, 2009.
- [21] R. Pande-Chhetri and A. Abd-Elrahman, "De-striping hyperspectral imagery using wavelet transform and adaptive frequency domain filtering," *ISPRS J. Photogramm. Remote Sens.*, vol. 66, no. 5, pp. 620–636, Sep. 2011.
- [22] H.-S. Jung, J.-S. Won, M.-H. Kang, and Y.-W. Lee, "Detection and restoration of defective lines in the SPOT 4 SWIR band," *IEEE Trans. Image Process.*, vol. 19, no. 8, pp. 2143–2156, Aug. 2010.
- [23] J. Jia, Y. Wang, X. Cheng, L. Yuan, D. Zhao, Q. Ye, X. Zhuang, R. Shu, and J. Wang, "Destriping algorithms based on statistics and spatial filtering for visible-to-thermal infrared pushbroom hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 6, pp. 4077–4091, Jun. 2019.
- [24] H. Cui, P. Jia, G. Zhang, Y.-H. Jiang, L.-T. Li, J.-Y. Wang, and X.-Y. Hao, "Multiscale intensity propagation to remove multiplicative stripe noise from remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 4, pp. 2308–2323, Apr. 2020.
- [25] P. Rakwatin, W. Takeuchi, and Y. Yasuoka, "Stripe noise reduction in MODIS data by combining histogram matching with facet filter," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 6, pp. 1844–1856, Jun. 2007.
- [26] P. Rakwatin, W. Takeuchi, and Y. Yasuoka, "Restoration of Aqua MODIS band 6 using histogram matching and local least squares fitting," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 2, pp. 613–627, Feb. 2009.
- [27] B. Cao, Y. Du, D. Xu, H. Li, and Q. Liu, "An improved histogram matching algorithm for the removal of striping noise in optical remote sensing imagery," *Optik*, vol. 126, no. 23, pp. 4723–4730, Dec. 2015.
- [28] F. L. Gadallah, F. Csillag, and E. J. M. Smith, "Destriping multisensor imagery with moment matching," *Int. J. Remote Sens.*, vol. 21, no. 12, pp. 2505–2511, Jan. 2000.
- [29] B. Datt, T. R. McVicar, T. G. Van Niel, D. L. B. Jupp, and J. S. Pearlman, "Preprocessing EO-1 hyperion hyperspectral data to support the application of agricultural indexes," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 6, pp. 1246–1259, Jun. 2003.
- [30] L. Sun, R. Neville, K. Staenz, and H. P. White, "Automatic destriping of hyperion imagery based on spectral moment matching," *Can. J. Remote Sens.*, vol. 34, pp. 68–81, Jan. 2008.
- [31] X. Kuang, X. Sui, Q. Chen, and G. Gu, "Single infrared image stripe noise removal using deep convolutional networks," *IEEE Photon. J.*, vol. 9, no. 4, pp. 1–13, Aug. 2017.
- [32] P. Xiao, Y. Guo, and P. Zhuang, "Removing stripe noise from infrared cloud images via deep convolutional networks," *IEEE Photon. J.*, vol. 10, no. 4, pp. 1–14, Aug. 2018.
- [33] Y. Chang, L. Yan, H. Fang, S. Zhong, and W. Liao, "HSI-DeNet: Hyperspectral image restoration via convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 667–682, Feb. 2019.
- [34] Q. Yuan, Q. Zhang, J. Li, H. Shen, and L. Zhang, "Hyperspectral image denoising employing a spatial–spectral deep residual convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 1205–1218, Feb. 2019.
- [35] J. Guan, R. Lai, and A. Xiong, "Wavelet deep neural network for stripe noise removal," *IEEE Access*, vol. 7, pp. 44544–44554, 2019.
- [36] Y. Chang, L. Yan, L. Liu, H. Fang, and S. Zhong, "Infrared aerothermal nonuniform correction via deep multiscale residual network," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 7, pp. 1120–1124, Jul. 2019.
- [37] Q. Zhang, Q. Yuan, J. Li, X. Liu, H. Shen, and L. Zhang, "Hybrid noise removal in hyperspectral imagery with a spatial–spectral gradient network," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 7317–7329, Oct. 2019.
- [38] Y. Zhong, W. Li, X. Wang, S. Jin, and L. Zhang, "Satellite-ground integrated destriping network: A new perspective for EO-1 hyperion and Chinese hyperspectral satellite datasets," *Remote Sens. Environ.*, vol. 237, Feb. 2020, Art. no. 111416.
- [39] Y. Chang, M. Chen, L. Yan, X.-L. Zhao, Y. Li, and S. Zhong, "Toward universal stripe removal via wavelet-based deep convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 4, pp. 2880–2897, Apr. 2020.
- [40] W. Sun, K. Ren, X. Meng, G. Yang, C. Xiao, J. Peng, and J. Huang, "MLR-DBPN: A multi-scale low rank deep back projection fusion network for anti-noise hyperspectral and multispectral image fusion," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5522914.
- [41] S. Cao, H. Fang, L. Chen, W. Zhang, Y. Chang, and L. Yan, "Robust blind deblurring under stripe noise for remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5412117.
- [42] J. Guan, R. Lai, H. Li, Y. Yang, and L. Gu, "DnRCNN: Deep recurrent convolutional neural network for HSI destriping," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 7, pp. 1–14, Jan. 2022.
- [43] J. Song, J.-H. Jeong, D.-S. Park, H.-H. Kim, D.-C. Seo, and J. C. Ye, "Unsupervised denoising for satellite imagery using wavelet directional CycleGAN," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 8, pp. 6823–6839, Aug. 2021.
- [44] C. Wang, M. Xu, Y. Jiang, G. Zhang, H. Cui, L. Li, and D. Li, "Translation-SNet: A semisupervised hyperspectral image stripe noise removal based on transformer and CNN," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5533114.
- [45] E. Pan, Y. Ma, X. Mei, F. Fan, J. Huang, and J. Ma, "Progressive hyperspectral image destriping with an adaptive frequenial focus," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5517312.
- [46] N. C. Thompson, K. Greenewald, K. Lee, and G. F. Manso, "Deep learning's diminishing returns: The cost of improvement is becoming unsustainable," *IEEE Spectr.*, vol. 58, no. 10, pp. 50–55, Oct. 2021.
- [47] H. Shen and L. Zhang, "A MAP-based algorithm for destriping and inpainting of remotely sensed images," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 5, pp. 1492–1502, May 2009.
- [48] M. Bouali and S. Ladjal, "Toward optimal destriping of MODIS data using a unidirectional variational model," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 8, pp. 2924–2935, Aug. 2011.
- [49] Y. Chang, H. Fang, L. Yan, and H. Liu, "Robust destriping method with unidirectional total variation and framelet regularization," *Opt. Exp.*, vol. 21, no. 20, p. 23307, Sep. 2013.
- [50] Y. Chang, L. Yan, H. Fang, and C. Luo, "Anisotropic spectral–spatial total variation model for multispectral remote sensing image destriping," *IEEE Trans. Image Process.*, vol. 24, no. 6, pp. 1852–1866, Jun. 2015.
- [51] X. Liu, H. Shen, Q. Yuan, X. Lu, and C. Zhou, "A universal destriping framework combining 1-D and 2-D variational optimization methods," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 808–822, Feb. 2018.
- [52] X. Liu, X. Lu, H. Shen, Q. Yuan, Y. Jiao, and L. Zhang, "Stripe noise separation and removal in remote sensing images by consideration of the global sparsity and local variational properties," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 5, pp. 3049–3060, May 2016.
- [53] Y. Chen, T.-Z. Huang, L.-J. Deng, X.-L. Zhao, and M. Wang, "Group sparsity based regularization model for remote sensing image stripe noise removal," *Neurocomputing*, vol. 267, pp. 95–106, Dec. 2017.
- [54] H.-X. Dou, T.-Z. Huang, L.-J. Deng, X.-L. Zhao, and J. Huang, "Directional ℓ_0 sparse modeling for image stripe noise removal," *Remote Sens.*, vol. 10, no. 3, p. 361, Feb. 2018.

- [55] J.-L. Wang, T.-Z. Huang, X.-L. Zhao, J. Huang, T.-H. Ma, and Y.-B. Zheng, “Reweighted block sparsity regularization for remote sensing images desriping,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 12, pp. 4951–4963, Dec. 2019.
- [56] L. Song and H. Huang, “Simultaneous desriping and image denoising using a nonparametric model with the EM algorithm,” *IEEE Trans. Image Process.*, vol. 32, pp. 1065–1077, 2023.
- [57] N. Acito, M. Diani, and G. Corsini, “Subspace-based striping noise reduction in hyperspectral images,” *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 4, pp. 1325–1342, Apr. 2011.
- [58] X. Lu, Y. Wang, and Y. Yuan, “Graph-regularized low-rank representation for desriping of hyperspectral images,” *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 7, pp. 4009–4018, Jul. 2013.
- [59] H. Zhang, W. He, L. Zhang, H. Shen, and Q. Yuan, “Hyperspectral image restoration using low-rank matrix recovery,” *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 8, pp. 4729–4743, Aug. 2014.
- [60] Y. Chang, L. Yan, T. Wu, and S. Zhong, “Remote sensing image stripe noise removal: From image decomposition perspective,” *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7018–7031, Dec. 2016.
- [61] Y. Chang, L. Yan, and S. Zhong, “Transformed low-rank model for line pattern noise removal,” in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1735–1743.
- [62] W. Cao, Y. Chang, G. Han, and J. Li, “Desriping remote sensing image via low-rank approximation and nonlocal total variation,” *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 6, pp. 848–852, Jun. 2018.
- [63] J.-H. Yang, X.-L. Zhao, T.-H. Ma, Y. Chen, T.-Z. Huang, and M. Ding, “Remote sensing images desriping using unidirectional hybrid total variation and nonconvex low-rank regularization,” *J. Comput. Appl. Math.*, vol. 363, pp. 124–144, Jan. 2020.
- [64] H. Zhang, J. Cai, W. He, H. Shen, and L. Zhang, “Double low-rank matrix decomposition for hyperspectral image denoising and desriping,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5502619.
- [65] S. Zhao, Q. Yuan, J. Li, Y. Hu, X. Liu, and L. Zhang, “A fast and effective irregular stripe removal method for moon mineralogy mapper (M^3),” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4600119.
- [66] K. Naganuma and S. Ono, “A general desriping framework for remote sensing images using flatness constraint,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5525016.
- [67] X. Su, Z. Zhang, and F. Yang, “Fast hyperspectral image denoising and desriping method based on graph Laplacian regularization,” *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5511214.
- [68] G. Xu, L. Deng, H. Wang, and H. Zhu, “CPAD: Component pixel-aware multispectral image desriping,” *IEEE Trans. Aerosp. Electron. Syst.*, early access, Jun. 13, 2023, doi: [10.1109/TAES.2023.3285195](https://doi.org/10.1109/TAES.2023.3285195).
- [69] Q. Xie, Q. Zhao, D. Meng, Z. Xu, S. Gu, W. Zuo, and L. Zhang, “Multispectral images denoising by intrinsic tensor sparsity regularization,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1692–1700.
- [70] H. Fan, Y. Chen, Y. Guo, H. Zhang, and G. Kuang, “Hyperspectral image restoration using low-rank tensor recovery,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 10, pp. 4589–4604, Oct. 2017.
- [71] Y. Chen, T.-Z. Huang, and X.-L. Zhao, “Desriping of multispectral remote sensing image using low-rank tensor decomposition,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 12, pp. 4950–4967, Dec. 2018.
- [72] W. Cao, K. Wang, G. Han, J. Yao, and A. Cichocki, “A robust PCA approach with noise structure learning and spatial-spectral low-rank modeling for hyperspectral image restoration,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 10, pp. 3863–3879, Oct. 2018.
- [73] Y. Wang, J. Peng, Q. Zhao, Y. Leung, X.-L. Zhao, and D. Meng, “Hyperspectral image restoration via total variation regularized low-rank tensor decomposition,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 4, pp. 1227–1243, Apr. 2018.
- [74] H. Fan, C. Li, Y. Guo, G. Kuang, and J. Ma, “Spatial-spectral total variation regularized low-rank tensor decomposition for hyperspectral image denoising,” *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 10, pp. 6196–6213, Oct. 2018.
- [75] Y. Chen, W. He, N. Yokoya, and T.-Z. Huang, “Hyperspectral image restoration using weighted group sparsity-regularized low-rank tensor decomposition,” *IEEE Trans. Cybern.*, vol. 50, no. 8, pp. 3556–3570, Aug. 2020.
- [76] Y. Chang, L. Yan, X.-L. Zhao, H. Fang, Z. Zhang, and S. Zhong, “Weighted low-rank tensor recovery for hyperspectral image restoration,” *IEEE Trans. Cybern.*, vol. 50, no. 11, pp. 4558–4572, Nov. 2020.
- [77] Y.-Y. Liu, X.-L. Zhao, Y.-B. Zheng, T.-H. Ma, and H. Zhang, “Hyperspectral image restoration by tensor fibered rank constrained optimization and plug-and-play regularization,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5500717.
- [78] J. Cai, W. He, and H. Zhang, “Anisotropic spatial-spectral total variation regularized double low-rank approximation for HSI denoising and desriping,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5536619.
- [79] S. P. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2010.
- [80] Y. E. Nesterov, “A method of solving a convex programming problem with convergence rate $O(\frac{1}{k^2})$,” *Dokl. Akad. Nauk SSSR*, vol. 269, no. 3, pp. 543–547, 1983.
- [81] D. L. Donoho, “De-noising by soft-thresholding,” *IEEE Trans. Inf. Theory*, vol. 41, no. 3, pp. 613–627, May 1995.
- [82] W. Deng, W. Yin, and Y. Zhang, “Group sparse optimization by alternating direction method,” *Proc. SPIE*, vol. 8858, Sep. 2013, Art. no. 88580R.
- [83] M. K. Ng, R. H. Chan, and W.-C. Tang, “A fast algorithm for deblurring models with Neumann boundary conditions,” *SIAM J. Sci. Comput.*, vol. 21, no. 3, pp. 851–866, Jan. 1999.

• • •